Stony Brook University

Multi-source Image Integration Towards Solar Forecast

A Dissertation presented

by

Zhenzhou Peng

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Computer Engineering

Stony Brook University

May 2016

Stony Brook University

The Graduate School

Zhenzhou Peng

We, the dissertation committe for the above candidate for the

Doctor of Philosophy degree, hereby recommend

acceptance of this dissertation

Dantong Yu - Dissertation Advisor
Adjunct Professor, Department of Electrical & Computer Engineering

Alex Doboli - Chairperson of Defense
Professor, Department of Electrical & Computer Engineering

Yue Zhao
Assistant Professor, Department of Electrical & Computer Engineering

Dong Huang
Lead Research Scientist, NASA Goddard Flight Service Center

This dissertation is accepted by the Graduate School

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

Multi-source Image Integration Towards Solar Forecast

by

Zhenzhou Peng

Doctor of Philosophy

in

Computer Engineering

Stony Brook University

2016

Accurate prediction of solar energy in short-term and mid-term horizons becomes increasingly important for harvesting solar energy and improving its viability in comparing with fossil fuels. Because clouds are the primary cause of large fluctuations in solar radiation, estimating cloud movements and correlating cloud activities and the variability of solar irradiance are the essential components in short-term or mid-term solar predictions and play a vital role in the subsequent mitigation in responding to potential solar power fluctuations. However cloud detecting and tracking is very challenging due to the volatility and complexity of clouds and expensive meteorological instruments or remote sensing technologies (e.g., satellite imagery) that often have insufficient resolu-

tions and limited availability. Ground-based sky imager offers higher temporal and spatio-resolution than does satellite. However it has a small field of view and lacks of spatial information of clouds, and solar forecast systems utilizing this type of imager are still inefficient and ineffective in cloud identification, motion tracking and as well as short-term irradiance prediction.

To fully address the problems of cloud detection and tracking and construct robust solar forecast system, we focus on three major thrusts: robust cloud motion tracking, multi-camera integration, and multi-channel satellite utilization. First, to improve the accuracy and robustness of cloud motion estimation, we designed a hybrid cloud tracking model to incorporate the strength of multiple classic techniques of motion estimation. This new model employ block-wise motion estimation, extracts the dominant motion patterns via histogram statistics. Furthermore, it estimates a dense motion field at a pixel level via customized motion filters and a refined objective function. Compared with state-of-the-art methods, our new model achieved at least 30% and 10% reductions to the errors of motion estimation respectively in simulated and real cloud datasets. Second, to overcome the limitations of ground-based sky images, we implemented a system to integrate multiple sky imagers. The system utilizes a novel method of identifying and tracking clouds in three-dimensional space and constructs an innovative pipeline for forecasting surface solar irradiance based on the image features from clouds. As a result, it robustly detected clouds in multiple layers and outperformed state-of-the-art methods by at least 26% accuracy improvement. Third, to extend the forecast horizons, we devised a mid-term forecast system that integrates the multi-channels of geostationary satellite images and adopts the optical flow approach to estimate cloud motions from noisy satellite images and better subsequent extraction of local cloud features. It employs a robust regression model to combine multiple features together to eliminate outliers and reduce prediction error. The resulting system demonstrates significant improvements over the baseline model in predicting solar irradiance.

For my beloved family, advisor, friends and colleagues.

Contents

List of Figures

List of Tables

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Prof. Dantong Yu for the continuous support of my Ph.D study and related research, for his patience, motivation, and numerous pieces of priceless advice. His guidance helped me in all the time of research and writing of this thesis.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Alex Doboli, Prof. Yue Zhao, and Dr. Dong Huang, not only for their detailed comments and encouragement, but also for the insightful questions which incented me to widen my research from various perspectives.

My sincere thanks also goes to Prof. Sue Ellen Haupt, Paul D. Kalb, and John H. Heiser, who provided me an opportunity to join their research team, and who gave access to the Brookhaven National Laboratory (BNL) and research facilities. Without they precious support it would not be possible to conduct this research.

During my Ph.D study, I feel honored to work with a collection of colleagues at Stony Brook University and Computational Science Center in BNL. This includes Dr. Shinjae Yoo, Dr. Hao Huang, Shuchu Han, Jin Xu, Dr. Yufei Ren, Dr. Tan Li, Li Shi, Dr. Shun Yao, and Cheng Chang. I want to express my grateful thanks to all of you, for all the joyful gathering and sincere support. I could not imagine a better work environment than I've found with all of you.

Last but not the least, I would like to extend special thanks to my dear family: my wife, my sister, and my parents for supporting and encouraging me spiritually throughout writing this thesis and my life in general.

Publications

- Zhenzhou Peng, Dantong Yu, Dong Huang, John Heiser, Shinjae Yoo, Paul Kalb, "3D cloud detection and tracking system for solar forecast using multiple sky imagers", *Solar Energy*, vol.118, pp. 496-512, Aug. 2015.

- Zhenzhou Peng, Dantong Yu, Dong Huang, John Heiser, Paul Kalb, "A Hybrid Approach to Estimate the Complex Motions of Clouds in Sky Images", *Solar Energy*, *under review*.

- Hao Huang, Jin Xu, Zhenzhou Peng, Shinjae Yoo, Dantong Yu, Dong Huang, Hong Qin, "Cloud motion estimation for short term solar irradiation prediction", in *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, IEEE, 2013, pp. 696-701.

- Zhenzhou Peng, Shinjae Yoo, Dantong Yu, Dong Huang, "Solar irradiance forecast system based on geostationary satellite", in *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, IEEE, 2013, pp. 708-713.

- Zhenzhou Peng, Shinjae Yoo, Dantong Yu, Dong Huang, Paul Kalb, John Heiser, "3D cloud detection and tracking for solar forecast using multiple sky imagers", in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, ACM, pp. 512-517.

Chapter 1

Introduction

Solar energy is one of the most promising renewable energy solutions, i.e., it is quieter and subjected to fewer installation restrictions, incurs lower maintenance cost, and has longer life time than other alternatives. Because of innovative solar technologies in photovoltaics (PV) plants and the increasingly competitive prices, solar-based electricity generators gain more interests in energy market than ever before. An ever-growing number of distributed roof-top solar panels and utility-scale solar plants and their associated energy storage in smart-grids or micro-grid operate in a high temporal- and spatio-resolution, and so engenders a pressing need for forecasting solar intermittency over a series of timescales. As cloud is the leading factor in ramp events of solar irradiance, existing solar forecast systems attempt to utilize various types of images to measure and track cloud information. However, clouds have complex variability and conditions, current image-based cloud tracking and solar forecasting are ineffective in representing true cloud movements and accurately predicting variations in solar irradiance.

To tackle these research challenges and improve the accuracy of solar forecasts in various temporal/spatial resolutions, we focus on building and enhancing image-based solar forecast system that can integrate multi-source images. In this thesis, we investigate and design a new motion estimation framework to

accommodate cloud tracking models and to detect motion field in various images. To enlarge the field of view (FOV) beyond the maximum range of individual camera and improve robustness of image-based solar forecast, we introduce a new multi-camera network and tracked clouds in a three-dimensional space. Furthermore, we design an innovative mid-term solar forecast system that is based on multi-channel satellite images and takes advantage a much larger FOV than that of any ground-based solution.

1.1 Research Motivations

Fossil fuels are still dominating the global energy sector. They are non-renewable and depletable eventually. Over-consumption of fossil fuel emits an excessive amount of air pollution and green house gases, and causes serious environmental issues and global warming. With these concerns with fossil fuels,renewable energy sources, on the other hand, became necessary for sustainable Earth and reached a 22.8% of market share in the global electricity sector [1]. In recent years, solar energy underwent the most rapid growth and gained the largest increase in renewable energy market. Solar energy has been the most abundant and cleanest energy solution. Since the invention of first modern solar cell in 1954 [2], two techniques, i.e., photovoltaic (PV) systems and concentrated solar power, have developed rapidly, and become increasingly cost-effective. In today's energy market, unsubsidized solar PV-generated electricity becomes even more competitive than fossil fuels and other energy sources in a growing number of locations around the World. Therefore solar PV starts to play a substantial role in global electricity market. This trend is confirmed by the fact that rooftop and utility-scale PV systems have been installed across the country at an unprecedented rate. Figure1-1 shows that U.S. market added a capacity of 7260 megawatt (MW) from the annual solar PV installation, and reached a total of 27.4 gigawatt (GW) solar capacity.

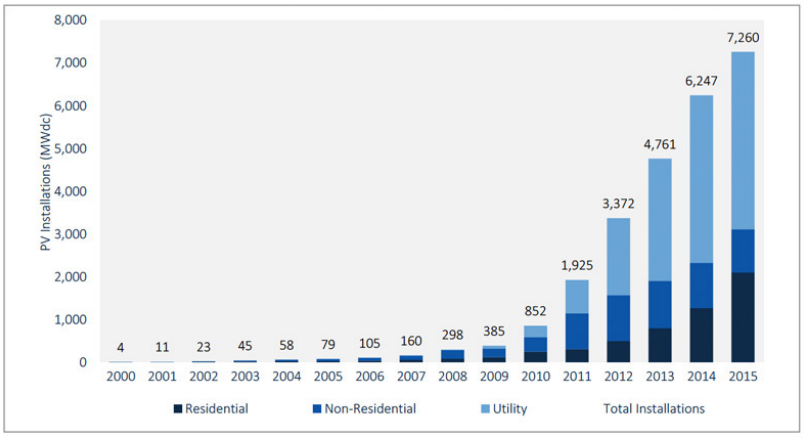


Figure 1-1: Annual U.S. Solar PV Installation, 2000-2015 (Figure 1.1 in Solar Market Insight 2015 Q4 [3]).

1.1.1 Emerging Market of Solar Energy

There are a variety of approaches to harvest solar energy in city and urban environments. Small scale applications for rooftops of residential and commercial buildings include solar thermal collectors and PV generators that integrate solar generated power into home electrical system or power grid for heating and illumination. Utility scale installation in scale solar power plants consists of thousands of PV arrays that convert sunlight directly to electric current or reflection mirrors that concentrate sunlight to boil fluid and turn the turbines of generators, and increases the capacity of energy market. To integrate the capacity of solar power plants into the traditional utility grids or compensate its volatility with operating reserves and energy storage, plant and grid operators often make their operation plans in advance, for example, minutes, hours and days ahead.

1.1.2 Intermittency in Solar Power Production

High variability and uncertainty of solar radiation compromise the reliability of solar energy production and slow down the pace of adopting solar power by utility companies [4]. The output of a solar power system varies significantly

between its ceiling and floor capacities in a matter of minutes or even seconds. Therefore, intermittency of solar power has become the biggest concern in enabling high penetrations of solar photovoltaics into the power grid and affected the stability and capacity of the power grid and its spinning reserve. In recent years, the emerging market of smart grid has also been driving the need of cost-effective management of power intermittency. Balancing the supply demand ratio and minimizing the operation cost of electric grid with the presence of sizable solar energy is critical to maximize the throughput of power generation and guarantee the quality of service for end customers. Depending on their particular solar applications, gird operators are concerned with foreseeing the fluctuations of PV production and designing responding strategies for various scenarios and in a variety of time resolutions.

Short-term solar irradiance fluctuations, usually in a sub-hour scale, lead to the same fluctuation in effected solar power plants. If left unmitigated, their unstable power output can subsequently cause voltage flicker and large voltage variations, and even trip automated line equipment that are attached to distribution feeders. The larger the solar generator's capacity is, the more maintenance and repair cost it incurs to the power utility. This type of intermittency often requires costly ancillary services to ramp up or spin down within a short time window. Long-term solar irradiance fluctuations and their causal affects to the variations of solar power are the key issue for the unit commitment by a large-scale power generator. To mitigate the long-term intermittency of solar power generation of a utility scale, plant operators must access to supplementary power sources when over-committing its production, and curtail solar energy when under-committing.

Among many disadvantages of solar energy, short-term variability within several minutes is the worst concern to grid operators in many regions. For example, Figure 1-2 shows a one-day worth of ramp events in term of change in Global Horizontal Irradiance (GHI) at Long Island Solar Farm (LISF) in 2014.

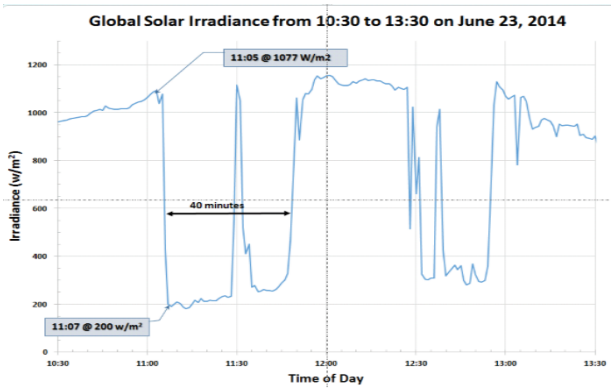


Figure 1-2: GHI Drop at Long Island Solar Farm (LISF).

GHI Drop	>100 W/m2	>200 W/m²	>300 W/m²	>400 W/m²	>500 W/m²
2012	306	259	210	173	122
2013	311	271	224	185	137

Figure 1-3: Annual report of GHI drop events at LISF.

The ground-level GHI reading frequently declined by more than 80% within 5 minutes. Such an intermittency of GHI occurs during one third of days in a year. Figure 1-3 presents the histogram of ramp events measured by the changes in GHI.

1.1.3 Need of Solar Forecasting

As the intermittency of solar resources is the biggest hurdle in solar energy utilization, recently, independent service operators and utility companies urgently need a cost-effective solution to manage, control, and mitigate the variability during solar power production. This is essential to the electricity value chain to balance supply-demand interaction, reduce the cost of operating electricity grids, and maximize the benefits of solar powers. Multiple decisions, such as, to charge and discharge energy storage, to dispatch spinning reserves, to incorporate production into in smart grids, and to mitigate the instability of solar power, require to accurately estimate solar variability in various timescales. As a result, solar forecasting plays an essential role to operating grids to accommo-

date power interrupts while maximizing the efficiency of PV system for peak demand. Accurate estimation of solar irradiance is an important aspect of optimizing the management strategy of solar power plants and achieving an optimal match between energy production and real-time consumption, and as a result, enables a smarter and more dynamic power grid than the current one.

Even all participants in electricity value chain show considerable interests in solar forecasting, there is no industrial standard available. Numerous ad-hoc solutions were developed to address diverse technical requirements of forecast outputs, quality and timescales. We borrow the classification mechanism used by atmospheric research to group Solar forecast systems into four categories according to their spatial scales: microscale, mesoscale, macroscale and megascale, and three classes based on their different forecast horizons in the time dimension: short-, mid-, and long-term [5] [6]. Each of them is useful for a specific application of utilizing solar energy.

Short-term forecast system predicts solar activities minutes ahead in a local scale, is important for a number of applications, for example, smart storage in microgrids, roof-top solar panels, and power grids, and provides the future knowledge of shaded/nonshaded areas in large-scale PV system for maximizing power. New advances in solar forecasting research include high spatial resolution and improved accuracy, and help assess the variability in a low cost and predict short-term ramp events in real-time or near real-time. Mid-term forecast targets at a larger area (meso- and macro-scale) than does short-term one, and has the time range between 20 minutes and up to 6 hours. This type of forecast is also useful to cover large-scale PV system and predict the load variations caused by solar resource intermittency. The mid-term information can be integrated into a strategy system to balance load requirements and supply, i.e., ramping up spinning reserves to compensate the generation slack, and curtailing solar energy to shed excessive supply in a hybrid grid. Recently, "nowcasting" is often used to represent the combination of microscale and mesoscale forecasts, and referred

as short-range predictions between minutes and up to six hours [7]. Long-term forecast leverages the numerical weather prediction (NWP) model and handles low-resolution data in a macroscale or larger. It focuses on assessing long-term solar variability, and serves the day-ahead energy market. Unit commitment under volatile production of renewable energy requires. Accurate hours and days-ahead predictions are indispensable to the unit commitment of unstable solar energy that is greatly affected by environment [8] and help integrate solar power into modern power grids, balance heterogeneous power sources, design efficient storage systems, and make optimal purchase decision.

1.1.4 Image-based Solar Forecast

Cloud is the primary source of large fluctuations in solar radiation at commercial solar farms. The main interaction between cloud and radiation in visible spectrum is scattering that can rapidly change gross irradiance as measured by surface radiation sensors. Cloud-induced solar irradiance variability has become one of the biggest concerns in power grid as the market share of solar energy has ramped up steadily in recent years, a.k.a, solar energy penetration. To predict solar irradiance ramps essentially depends on cloud observations and predictions of various cloud characteristics, e.g. cloud cover, sun-occlusion effects, cloud movements and cloud changes. Consequently, the capability to predict the presences of clouds and extract their relevant features is critical for estimating the variation in solar energy and thereby for mitigating the effects of the output fluctuations in utility-scale PV power plants. To accurately determine the properties of local clouds, it is common to utilize measurements from ground-based instruments, such as cloud radar and LIDAR. However, because of their high cost and limited availability, these tools are not widely used for localized, short-term forecasts. Although ground-based reference sensor network is able to predict very-short term solar forecast [9], additional hardware support is needed for

tracking over a large region with longer time horizons. With the advent of inexpensive digital cameras and satellite imaging techniques, clouds are visualized and located for inferring their optical properties and estimating their movements in a short time window. For local scale applications of short-term prediction, digital cameras deployed at ground level provide real-time or near real-time image streams to capture the movement of clouds and optical properties that are within the camera field of view. Whereas for the large scale applications, such as solar farm, satellite images are useful to monitor the cloud condition of a certain region and track cloud motion hours-ahead. Equipped with various types of cloud tracking device, we can monitor and track clouds across different timescales and extract various non-intuitive information, including height and velocity [10] [11].

1.2 Research Challenges

Using various imaging techniques is a promising technique to represent and visualize clouds' properties, e.g. optical depth and velocity with a rich set of image features at pixel level, and ultimately track clouds. However to regress cloud images (as regression inputs) to high-accuracy solar forecast (as regression outputs) is not a trivial task. In addition to the physical limitations, such as camera's field of view and resolutions, there are challenges with image-based approaches that must handle the complexity of cloud physics, extract relevant features, and create models for cloud activities.

1.2.1 Cloud Detection In Image

Cloud tracking and property extraction is well studied in climate and atmospheric research. Due to the limitation of availability and cost of meteorological devices, the majority of research outcome can not be directly applied to solar forecast. Many inexpensive image-based cloud observation systems were de-

veloped to address the well recognized requirements of collecting and assessing cloud characteristics. However, challenges are encountered even for a simple task of identifying clouds at the pixel level of an image. Because clouds have distinct properties, such as different optical depths and brightness variations, cloud images have various distributions in RGB channels and a wide range of brightnesses in different sky scenes or image frames. In particular, certain optically thin clouds may have similar textures and colors to those of the background sky pixels, whereas some clouds may appear both dark and white within the same sky image. Therefore, it is difficult to identify a fixed threshold to separate clouds from sky. For instance, cirrus clouds are barely distinguishable from the clear-sky background in TSI images [12]. Moreover, under different weather conditions and at different solar angles, the clouds presented on a bitmap image may appear to have various brightnesses and a large range of intensities. Therefore, a robust methodology is required to assess sophisticated image-based features to capture the pixel-wise differences or the regional textural differences between cloudy and clear regions.

Most of prior works try to detect cloud pixels by utilizing thresholds of prominent features, e.g. RGB or red-blue-ratio (RBR), at pixel level to separate clouds from clear sky regions [13, 14]. However the threshold-based methods are not robust enough for the cases with variations of lighting condition even in the same scene. To extract the accurate threshold to detect cloud pixels, training samples need to be selected manually to cover various sky scenes and cloud/weather conditions. Even with the design of smart adaptive method to determine a flexible threshold [15], accuracy is often compromised in the extreme cases that in fact are common in daily observations. It is very challenging to automate the threshold-based method and serve the purpose of cloud detection under different cases.

1.2.2 Vertical Distribution of Clouds

To research on cloud-induced variations of solar irradiance, clouds must be clearly identified in various types of images, and more importantly located in three dimensional space to provide faithful spatial correlation with ground-based measurements. However, due to various vertical distribution, clouds generally have different layers with base height ranging from hundreds of meters to several kilometers. More importantly, multiple layers of clouds have textural distinctness and various wind field. Therefore, sun-occlusions effects are challenging to be tracked due to various layers and multiple movements. It is critical to estimation of ramps of solar irradiance especially when multiple layers of clouds co-influence the solar energy reception. For a large-scale application, such as solar array or power grid network, multi-layer clouds, especially the overlapping areas, lead to instability of production that are difficult to track.

The core aspect of multi-layer cloud identification is to determine various cloud motions and the cloud height on images. To achieve this goal, the cloud layers on a single image source are normally determined based on the smooth layers in motion segmentation [16] that is often inaccurate and erroneous in non-rigid tracking. With integration of multiple image sources, we can apply different mechanism for layers determination with a higher accuracy than that of the smooth layers of motion segmentation. With multiple sky cameras, clouds can be reconstructed and identified in a 3-dimensional space. Combined with multiple satellite channels, each of which has different spectral sensitivity, we can explore to use cloud textural patterns to separate vertical layers.

1.2.3 Complex Cloud Types and Movement

Though cloud motions can be derived from weather forecast models several days or hours ahead, the spatial and temporal resolution of the result is low and can not be assimilated directly into solar forecast system to capture irradiance

ramps. With the advance of visualization techniques, movement of clouds can be detected in various types of images, e.g. visual channel of satellite imager and sky cameras at ground level. Aided by vertical distributions of clouds, we can design algorithms to accurately convert pixel-wise motion vector to real cloud movements in spatial domain and use the results for the short-term prediction of sun-occlusion events. However, due to the complexity of clouds and non-rigidity of their movements, cloud motion tracking is a challenging problem that has not been fully addressed.

The first issue is the variations of clouds' type and color in a stream of cloud images. It is extremely difficult to treat clouds as objects and detect their shifts and transformations between frames based on image segmentation [17] because observed clouds have various shapes and classifications. Even for the same type of clouds, with different weather, lighting conditions or viewing angle, they may acquire distinct color representations in cloud images and are hard to be identified as "relevant" to each other. Even the traditional motion tracking methods can incorporate image segmentation, object detection, and edge information retrieval, they tend to generate inconsistent results and need to be tuned carefully for different images and cloud conditions. In practice, tracking models are customized and vary with different cameras in use and resolutions in field of view. None of prior approaches took cloud information into account and adjusted automatically for different conditions.

Another factor that impairs the accuracy of cloud tracking is the deformation and arbitrary shape change of clouds. Even many techniques were proposed based on computer vision research, existing models tend to over-simplify the tracking problem and assume constancy among clouds and their velocity over time [18]. These methods work well for scatter clouds that only have translative shift, and involve moderate shape changes and vertical shifts. However, when dealing with the complex cases, such as significant cloud deformations and multiple layers of different motions, they suffered the loss of accuracy [19,20]. Even

with the assistance of post-processing techniques, e.g. clustering and anomaly detection, they only attained limited improvement [12, 21].

1.2.4 Multi-source Data with Various Temporal and Spatial Scales

Long-term solar forecasting model takes advantage of the best numerical weather prediction and integrates big data from various types of observations and measurements. With the emerging need of cloud prediction in other time horizons, to determine the properties of clouds and estimate cloud movements in various temporal and spatial resolutions has become increasingly pervasive in recent studies.

Ground-based instruments, measurement stations, and remote sensing techniques can improve solar irradiance forecast. However, common meteorological instruments such as radar and LIDAR, because of their high cost and limited availability, are not widely utilized for localized and short-term forecasts. Although solar radiation is able to be predicted based upon ground-based reference sensor network with a rich set of sensor information [9], additional instruments still needs to be incorporated for tracking over a large region.

With the advent of inexpensive digital cameras, several types of imagers, e.g. total sky imager (TSI), whole-sky imagers, have become widely used as cost-effective tracking device for visualizing clouds and estimating cloud movements in a short time window. However these imagers have limited field of view and image resolutions, they only faithful acquire a sky view within several kilometers in radius and fit very short-term solar forecasts of up to fifteen minutes. The field of view of an individual camera is insufficient for large scale solar farms consisting of millions of PV panels. Even with multiple imagers that are distributed in several locations, their enlarged FOV still has constraints because individual cameras can not straightforwardly provide clouds' spatial informa-

tion, such as base height and vertical layers.

Contrasting to traditional sky imagers, many new systems have enlarged view range and provide detailed cloud patterns. One pervasive approach is to replace the existing cameras of sky imagers with a high definition (HD) web camera to obtain real-time video stream and high-resolution sky snapshots. The HD fish-eye camera not only has a widened view angle to effectively monitor clouds that are far away, but also enhances cloud features with clearer contours and shapes and richer textural patterns than those of traditional ones. However constrained by required computational power and real-time processing speed, existing works only consider a small number of frames with reduced resolution and field of view. None of existing forecast systems integrates multiple imagers synchronously.

Furthermore, to obtain a view of clouds in meso- and global-scale and forecast solar energy with hours ahead time horizon, we resort to satellite imager system. Prior approaches often rely on only the visible channel of satellite imager to generate cloud motion field and forecast solar energy. These single-channel methods, nevertheless, have high error rate due to the variant cloud shape and inconsistent cloud motion. It is also difficult to identify cloud and/or obtain its thickness only from the visible channel because it has coarse spatial resolution (1km x 1km per pixel) and fails to distinguish cloud types. In other words, cloud is even smaller than the pixel (grid) size of a satellite image, or the visible channel shows only a limited spectrum of clouds, and does not recognize the remaining ones. Though utilizing multiple channels of satellite imagers appears in previous studies, none of them attempted to combine data of various resolutions and assign weights to different spectrum channels for the purpose of robust cloud tracking and solar forecasts.

1.2.5 Solar Irradiance Modeling

Ultimately, we need to create a regression model that consists of a function in spatial and temporal dimensions, takes in significant features from cloud image and regresses toward the targeted solar irradiance changes or solar energy fluctuations as measured by ground-based Pyranometer or solar inverter and net meters. The most difficult task in this mechanism lies in a proper extraction of spectral and textural feature from clouds. In most cases, cloud objects appear to have volatile shape and various luminance patterns in their images. Moreover, different cloud types are distinct in terms of solar irradiance absorption and reflection. Consequently, more efforts are needed to determine potential effective cloud features from image, especially around the region that potentially blocks the Sun. Due to the instability that is caused by noisy pixels and luminance variation in images, the method needs to correctly distinguish local features and consider the smoothness of global features.

1.3 Research Contributions

Herein we propose an image-based solar forecast architecture that contains five components (See Figure 1-4). This architecture is adaptive to various types of images from multiple heterogeneous sources, such as ground-based imagers and satellite channels, and provides both short-term and mid-term solar forecasts corresponding to inputs. The first component in our reference implementation is cloud identification module that detects cloud pixel from input images. It is followed by a cloud motion tracking module that estimates cloud motion vectors based on the image processing model and the physics model. Subsequently, a layer determination model categorizes clouds and extracts the layer information at the pixel level. The fourth and fifth components focus on cloud prediction, feature extractions, solar irradiance regression. Prominent image features, such as cloud optical properties, along with detected motion vectors that are obtained

by our intelligent optimization algorithms, serve as key ingredients to the last two modules that apply machine learning methods to choose important features and regress to our prediction targets.

This architecture entails three objectives of this thesis work: Cloud detection and motion tracking, short-term forecast based on multi-TSI network, and mid-term forecast via multi-channel satellite image integration.

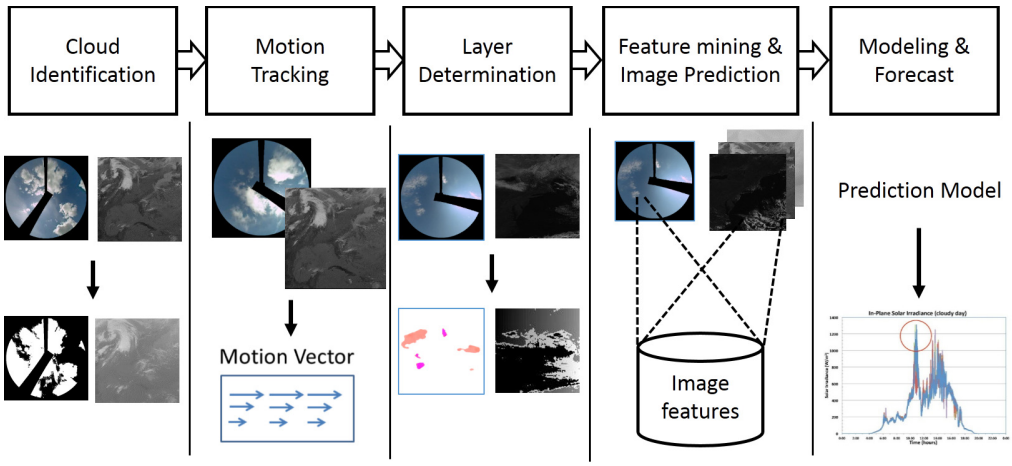


Figure 1-4: Image-based Solar Forecast System

1.3.1 Cloud Identification and Motion Tracking

To fully address the problem of tracking cloud motions and provide accurate prediction of sun-occlusion effect is essential to short-term solar forecast. In this dissertation, we first investigate previous works on cloud motion estimation based on sky images and summarize them into several categories. After analyzing the advantages and disadvantages of classic models, we then proposed a hybrid model to integrate previous approaches to form a robust motion tracking framework. It builds on top of block-matching technique to mitigate image noise and uses the global pattern of motion vectors to refine the field of cloud motion and remove outliers from detected motion vectors. In this new model, we designed a novel energy-like function with a penalty term to encourage clouds to follow the major trend of movements and at the same time allow

small motion perturbation to track shape deformation. To better optimize the tracking results, we iteratively solve the objective function and incorporate customized motion filters to improve the robustness when facing the outliers. At last, to better compare the new model with state-of-the-art methods, we collect images from various sites/image sources and build a simulation framework to synthesize cloud image with clouds' shape deformation and movements. The comparison results on both real and synthetic images confirmed that the hybrid model acquires the best performance of motion estimation and is adaptive to track cloud deformations.

1.3.2 Framework of Multiple Imager Integration

To enlarge the field of view of sky imagery and enable the extraction of three-dimensional information of clouds, in this dissertation, we deployed three cameras at various locations to obtain synchronous image streams of clouds from different view angles. We subsequently designed a short-term forecast framework to detect clouds' motion and layers from these image streams, and build solar irradiance models of using image-based features.

Different from prior studies, we proposed a supervised classifier to identify cloud pixels and devised local correction algorithm that utilizes three synchronous multiple TSIs to correct each TSI's bias or abnormal exposures. In order to improve the accuracy of motion estimation, we segmented cloud image based on cloud distribution and used 9-frame to estimate the base height and movement of clouds robustly in 3D space. To better accommodate the block matching algorithm for 3D tracking, we designed a new similarity function incorporating temporal and spatial correlation. Afterward, to integrate individual block into various layers, we developed a clustering-based technique to categorize block-wise tracking results into multiple cloud layers, each of which forms a major wind field. With the spatial information from multiple 2D images, we

were able to use sky images to generate image-based features at corresponding pixels, and investigated various regression-based models to undertake the short-term solar irradiance forecast. Within the framework of multi-TSI network, our proposed solar forecasting system has the ability of detecting and tracking clouds, predicting relevant image-based features, and as a result extend the forecast horizons from five minutes (the optimal prediction regime for a single TSI) to 15 minutes.

1.3.3 Multi-spectral Satellite Image Integration

Ground-based cameras have limited prediction horizon of up to half an hour for the East Coast of United States. To extend forecast horizons from sub hour to up to several hours as required by large-scale solar applications, we utilized satellite images to track clouds and model the changes in ground-level irradiance. Existing satellite-based cloud observation and solar forecasting systems mainly focus on single channel, e.g. visible channel or infrared channel, when extracting cloud properties and optical characteristics. In this dissertation, we present a mid-term solar forecast system that combines multi-channel satellite images for cloud tracking and feature extraction. To solve the challenges of satellite imagery, such as image noise and inaccurate timestamp, we designed a preprocessing pipeline to filter out outliers in its multi-channel dataset. Furthermore, we adopted an optical flow framework to recursively detect pixel-wise motion vectors following a coarse-to-fine scheme and attain robust motion tracking, which is distinct from prior works. Similar to the ground-based prediction system, we combines the motion field and image features from satellite image, and adopt regression-based models for mid-term prediction. To enhance the accuracy of forecast models, we incorporated the persistent model into our system and consequently generate hours-ahead forecasts of ground-level irradiance. Compared with state-of-the-art satellite-based system, our model performs sig-

nificantly better in detecting cloud motion vectors and forewarning ramp events.

1.4 Significance and Broader Impacts

Solar energy is distinct from conventional power in many aspects: generation, transition and commitment. With its growing penetration into national grid that was built for conventional power, accurate forecast of the true solar production level plays an essential role in ensuring the stable operation of power grid, and benefits all participants in the electricity value chain: solar power producers, utilities and independent system operator (ISO). This research will not only greatly improve the accuracy of solar forecast of various time scales, but also yield substantial benefits to many areas such as remote sensing, image-based cloud researches, and more importantly, other forms of renewable energy. (1) The intelligent integration of multi-source images can be applied in different research fields, for example, the framework of multiple ground-based imagers is beneficial to cloud modeling and atmospheric research, and supplies a cost-efficient alternative to the expensive LIDAR for observing clouds in 3D space. Furthermore, our framework supports the incorporation of heterogeneous images with different timescales and resolutions, i.e. ground-based and satellite images, and offers a great potential to explore many other hybrid models for understanding, modeling, and integrating renewable energy [7]. (2) Our research addresses the challenging problems in image-based solar forecast and proposes a generalized cost-effective architecture to modularize cloud detection/tracking and predict ramp events. Such an architecture can be extended to many solar forecast systems with various time scale and multiple image sources, and more importantly, is ideal to be incorporated into a parallel-computing system to efficiently and effectively undertake real-time solar forecasts for utility-scale solar system, e.g. Long Island Solar Farm (LISF). Consequently, our research enhances the efficiency, reliability, safety, and competitiveness of solar energy and

is valuable to the electricity value chain by helping build an elastic and adaptive energy pricing and commitment system.

1.5 Dissertation Overview

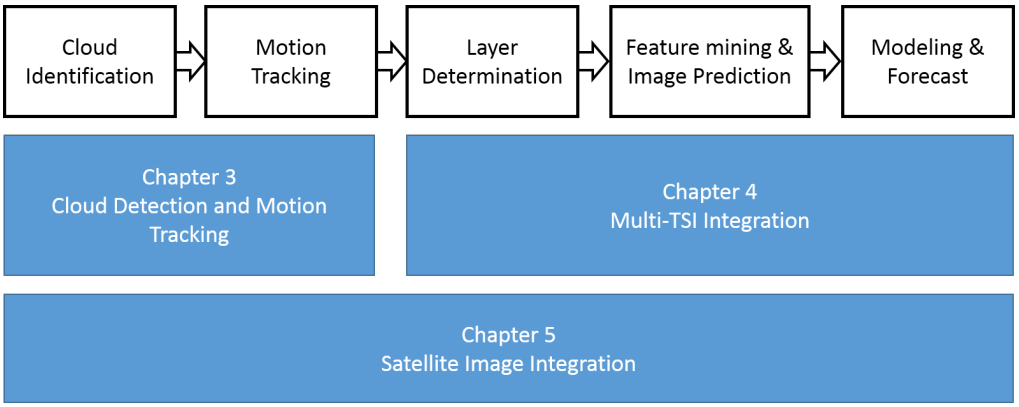


Figure 1-5: Structure of this dissertation.

This thesis starts with the introductory chapter that presents the overall motivations, challenges and our contributions and research significances in solar forecast. Chapter 2 introduces different types of sky imagery and presents a literature review on state-of-the-art methods on image-based cloud tracking and solar forecasting. The rest chapters in this dissertation focus on data integration in image-based solar forecast system. As shown in Figure 1-5, Chapter 3 provides the systematic studies in cloud identification and motion tracking in sky images and comprehensively evaluates classic models using both simulation data and the real images collected from various sites and/or sky cameras. In Chapter 4, we present a framework of local sensor network that is based on multiple cameras and utilizes spatial-temporal correlation between different cameras and/or consecutive timestamps. A new methodology is designed to derive 3D cloud information, detect multi-layer clouds using customized clustering algorithm, and at last, discuss feature extraction and robust regression models for the purpose of short-term forecast. Chapter 5 describes mid-term

forecast that incorporates the images from multiple spectrums of geostationary satellites. Furthermore, in this chapter, we detail a customized framework that preprocesses erroneous satellite image and ground measurements, undertakes cloud tracking in mesoscale, and extracts prominent features for robust forecast models. Lastly, Chapter 6 concludes our current work on utilizing multi-source image for short-term and mid-term solar forecast and proposes future works on data integration and modeling that are relevant to renewable energy.

Chapter 2

Background and Related Works

Image-based solar prediction system is widely adopted to capture the cloud activities that cause solar irradiance ramps and thereby accurately forecast the ramp events and intermittency in solar production. Their success attributes to cloud imaging with digital cameras that was proven to be effective in optical property extraction, cloud classification and motion estimation. In this chapter, we first present two types of imaging techniques (ground-based and satellite-based) that are widely applied in solar forecasting system, and survey various applications in the field of cloud property extraction from these two types of images. We then investigate several image-based motion tracking models that are pervasive in exploring complex cloud movements. Lastly, we introduce the classification of image-based solar forecast systems based on their temporal and spatial scales, and subsequently provide a literature review of solar forecast systems and applications in each class.

2.1 Ground-based Sky Camera

2.1.1 Hardware Development

To accurately determine the properties of clouds at local scale and within short time frame, it is common to utilize measurements from ground-based instruments, such as cloud radar and LIDAR. However, because of their high cost and limited availability, these tools are not widely used for solar forecasts of localized applications. Kleissl, et al. developed a cloud tracking system that only relies on sensor techniques, indirectly tracks clouds with a ground-based reference sensor network or the output of solar PV inverter in the format of time series data, and therefore requires no additional instruments [9]. This tracking system does not scale spatially. Additional hardware support is needed for tracking over a large region. An alternative is to use sky imager to photograph local view of sky and infer the irradiance properties of clouds and estimate their movements in a short time window [22].

One of the earliest sky cameras is Whole Sky Imager (WSI) [23]. It is developed to provide images of the whole sky dome in daytime for the purpose of cloud characterization, in particular thin cloud determination and starlight cloud decision, irradiance distribution analysis [24] [25]. However, due to the hardware cost and the resolution limitation within their effective FOV, WSI and its variants are not widely deployed for solar forecast in smart grids. Recent technology advances of digital security cameras, accompanied by their rapid price reduction rendered sky imagers cost-effective and high-resolution in their field of view.

Total Sky Imager (TSI) is the most pervasive sky camera for short-term solar forecast [26] [27]. It serves as a cost-effective alternative to identify the sophisticated correlation and causality among the real time information about cloud identification, location, speed, direction, and optical properties and the anticipated decrease or increase in power output at a specific centralized generating

(a) Overview

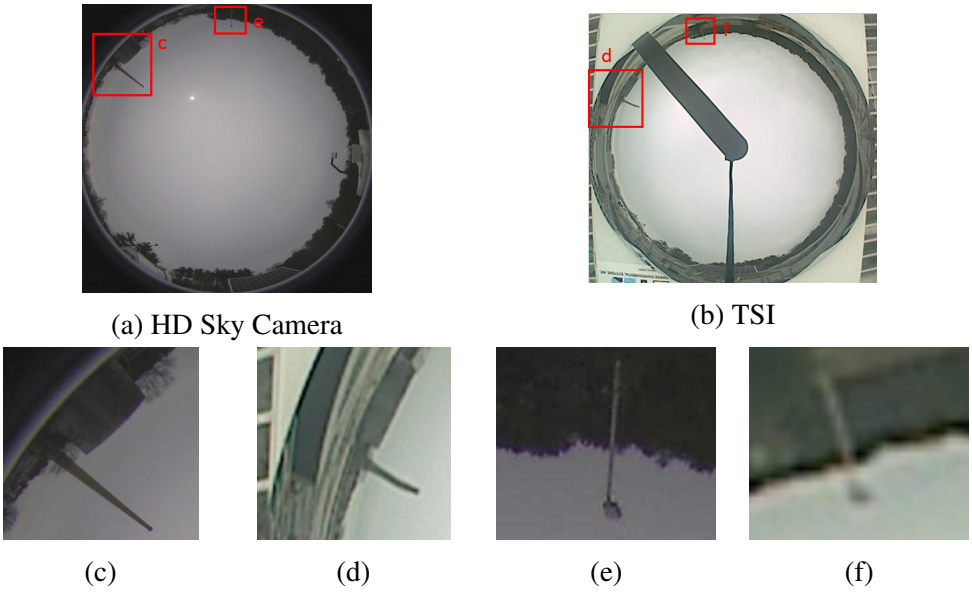
(b) Raw Image of TSI

(c) Undistorted View

Figure 2-1: Total Sky Imager (TSI)

site. TSI has been proved to be effective to provide continuous observations of clouds for short-term solar forecasts in minutes ahead [28]. As shown in 2-1, TSI adopts a built-in downward-facing camera and a dome-shape mirror to reflect sunlight from all angles above the ground level. A raw sky image is generated with a distorted view due to TSI's reflection mirror and difficult for tracking algorithms to obtain motion vectors. To ease the complexity of calculation in a non-linear dome space, we need to preprocess raw sky images, convert them into planar view via undistortion model and thereafter mask out obstacles that are inside the FOV of sky imager, but need to be marked as irrelevant pixels. To avoid excessive direct sunlights from overburning sensitive CCDs, TSI has a shadow band in its reflection mirror to block sun flare. Using TSI to capture clouds over short horizons has several shortcomings that might constraint TSI's ability and efficacy to detect and track clouds. TSI has one significant disadvantage where it adjusts the imaging settings in response to the overall scene brightness. Therefore, the raw TSI images captured under different lighting conditions lack inter-image consistency and often cannot faithfully represent the real view. Another issue with TSI is its built-in camera acquires only low-resolution images of up to 640 x 480. Moreover, because the TSI images are generated by reflections from a dome-shaped mirror, they typically suffer from the presence of ground obstacles within the FOV. More importantly, the shadow

band occludes approximately 14% of the total view [29], the camera-supporting arm are visible in the raw images and both need to be removed from images during the preprocessing step. A promising alternative approach to TSI is an inexpensive high-definition digital camera that is designed to overcome these issues on image quality. This approach was used successfully for short-term solar forecasts [30].



(a) HD Sky Camera (b) TSI

(c) (d) (e) (f)

Figure 2-2: Comparison of HD Sky Camera and TSI.

To enhance the quality of image of traditional sky imagers, researchers currently adopted new sky cameras with improved built-in camera to provide a better FOV and as well as HD resolution [31] [32]. Figure 2-2 shows that in contrast to TSI, the image quality of a HD sky camera significantly improves. Especially for the objects that are far away and have low zenith view angle, the HD camera still produces a clear image with less distortion than does the standard one. Therefore, HD camera is a promising tool for local scale cloud observation and tracking. However, HD images usually incur longer latency on processing and tracking than regular TSI images. Real-time prediction from HD images with stringent latency requirement necessitates the optimization of both algorithm design and parallel computation.

The majority of existing sky cameras/imagers are designed to image a hemispheric view of the sky. Their built-in cameras are either oriented upward, for direct image acquisition, or, downward, to capture the reflections from a spherical mirror [23] [27] [30] [29]. Other sky cameras also exist for various forms of cloud observations. One example is to use a laptop camera to collect image and track clouds in a mobile station. Another interesting example is to orient multiple sky cameras in low elevation angles to capture incoming clouds from horizon and extend the effective range of cloud tracking. Figure 2-3 shows a four-camera system that significantly increases the field of view and helps to observe far-away clouds. Combined with regular fish-eye cameras, this system can provide a much longer forecast horizon than TSI.

Figure 2-3: Four camera system with tilted view angle.

2.1.2 Cloud Observation based on Sky Imagery

Cloud observation via a ground-based sky camera plays an important role in recording and studying various weather phenomena. In recent years, different types of sky cameras have been developed and well utilized for automatic cloud observation and solar irradiance modelling [33]. In contrast to traditional measurements in meteorology, ground-based sky cameras directly record the localized view of sky as perceived by human eyes in their real-time image streams, demonstrate great potential in the field of cloud tracking and solar forecast. Compared with satellite imaging system and ground-based reference sensors, Sky imager is a promising tool for visualizing and tracking clouds in real time at local scales [26] [27] [34]. Recent work [30] shows that cloud coverage pre-

diction with HD camera has a small error rate of around 6% within one minute range and about 30% for 20 minutes interval.

The details of various cloud properties, such as optical depth, opaqueness and thickness,can be extracted from analyzing the red-green-blue (RGB) channels in sky images [35] [36] [37]. In addition to utilizing original RGB channels, Souza-Echer et al. proposed cloud detections in the hue-saturation-luminance (HSL) color space [38]. In our earlier work, we extracted generalized features from transformed luminance images [11]. Other approaches used the state-of-the-art classification methods, such as k-nearest neighbor(kNN) algorithms and binary decision tree, to determine cloud types based on the extracted textural features from sky imagery [39] [40]. Recent works [41] [42] [36] [43] have focused on exploring additional image characteristics or applying adaptive thresholds to improve the accuracy of classifying cloud types and identifying cloud pixels. In particular, the reported accuracy of the seven-type cloud classification ranged from 78% to 95% [43].

Sky imagery is also an effective tool for extracting rich cloud information via image processing techniques recording the movements of individual cloud, and demonstrates these superior capabilities that no other imaging system has. Based on the temporal correlations between consecutive frames of sky images, cloud movement can be obtained as motion vectors at the pixel level. One common approach adopted by previous researchers is to divide sky images into small regions/blocks and thereafter apply classic image processing models to obtain block-wise motion vectors [44] [45]. Kleissl et al. initially utilized the normalized cross-correlation method (CCM) to detect the velocity of regional cloud motion in sky images [46]. To improve the accuracy of cloud motion tracking, Huang et al. chose two different block sizes for block matching and undertook refined tracking based on the small block size within the range of the large block [21] [47]. To mitigate cloud deformation and motion perturbation, Huang et al. [48] explored an approach based on multi-frame motion back-tracking.

The cloud motion vector was then estimated based on the extrapolation of the known motion trend. In contrast to block-wise movement, the motions of clouds can also be represented in a scale smaller than block, e.g., individual pixels.

Some well-known computer vision techniques, such as Optical flow (OF) motion tracking, can be applied to estimate pixel-wise cloud motion. The OF method has the advantages of identifying the tiny difference between relevant pixels that are invisible to other methods and was successful in estimating even small movements of clouds [49]. By incorporating a deformation model, OF-based methods can capture large-scale cloud motions between consecutive frames based on image registration techniques [50]. In general, block-wise cloud tracking, as discussed in several previous works [10] [51], is an extension of the pixel-wise methodology. These two approaches fit different tracking conditions and are complementary to each other: on the one hand, block-based approach integrates the properties within a cloud block to mitigate the noise and misrepresentations of color in original images; on the other hand, the pixel-wise approach focuses on typical movements that occur at the pixel or sub-pixel scales.

Recently, sky imagery has been widely used in solar irradiance studies to predict the presence of clouds based on temporal correlations between frames. The core idea of this approach is predicting the cloud properties in and around the path between a radiation sensor and the Sun based on previously observed clouds and then estimating and regressing the cloud-induced fluctuations in irradiance based on both the predicted and observed properties [46]. By representing irradiance as a function of the features extracted from sky images, solar prediction can then be transformed into a problem of predicting future images and accurately resolving their corresponding cloud features [28] [11]. This methodology, as expected, relies on the ability to detect clouds and track their motion. To minimize the influence of erroneous motions, Huang et al. [48] added statistical features to the final irradiance forecast model. To further improve the accuracy of short-term solar forecasting, recent works have incorpo-

rated machine learning techniques into both cloud tracking and irradiance modeling [52] [47] [53].

2.2 Geostationary Satellite Imager System

Ground-based Sky cameras are effective to provide informative local measurements of clouds, however, have the limitation of application scale and as well as forecast horizons, i.e. several kilometers and up to thirty minutes ahead. Therefore, sky camera is usually deployed for certain point of interest, e.g. an individual Pyranometer or solar panel. For large scale of distributed solar panels, multiple cameras are needed which is costly and hard to validate the correctness of positioning sky cameras [11].

Remote sensing technique is complementary to the ground-based solution, extends the forecast horizon to mid-term, and provides a much larger field of view than that of TSI. Different from ground-based measurements, satellite captures the reflected radiation from the Earth and measures planetary albedo with its built-in radiometers. Satellite images, especially the composition images from multiple channels of geostationary satellites, are commonly adopted to estimate ground-level irradiance with both physical and statistical models [54] [55] [56] [57] [58]. Similarly, the approach for ground-based cameras is also applied to analyze the temporal correlations between consecutive frames of satellite imagery, and thereby becomes a promising tool to predict mid-term solar variations in a global scale [59] [60].

Existing approaches utilized visible channel of geostationary satellite (GOES) to study the solar irradiance and as well as track clouding [61] [10]. The satellite pixel-to-irradiance conversion error can be as low as 12-13% compared with ground measurements [62]. Therefore several state-of-the-art models are developed to obtain satellite-derived radiation data [63] [64].

2.3 Cloud Motion Tracking

Many motion-tracking techniques have been proposed to detect the motion of objects in terms of the pixel-wise movement across different images. However, only a few of them can be used to track cloud motion because of its non-rigid shape and formation. In general, the cloud motion vectors are important to numerical weather prediction models, and usually are obtained from satellite images [65] [66] [67] [68]. With the advent of inexpensive digital cameras and the emerging need for solar forecast with the fine granularity that is beyond the spatio-temporal resolution of satellite images, recent researches focused on using these ground-based cameras to track the very short-term motions of clouds [49] [48] [46]. These methods fall into three main categories based on the scale and tracking criterion of the motions.

2.3.1 Optical Flow Based Motion Tracking Methods

In the field of computer vision, motion tracking is usually resolved by estimating the Optical Flow (OF), i.e., the pixel-wise distribution of prominent velocities and directions of brightness/texture patterns on an image. In general, an OF method can obtain dense motion vectors at the granularity of a pixel, and was proven to be quite effective in detecting cloud motions in satellite images [68]. Horn and Schunck (HS) [69] and Lucas and Kanade (LK) [70] proposed the original approaches of optical flow. The HS model formulates optical flow as an optimization problem assuming that motions have global smoothness. The advantage of this model is that it can propagate information over a large distance within the image and fill texture-less regions, e.g. thin clouds, with the motions extracted from the boundaries. Consequently, it is widely incorporated into other variational models and customized for estimating cloud motions [19]. Another state-of-the-art methodology, i.e., LK, is to constrain motions and preserve local smoothness, i.e., satisfying the gradient constraint among neighbor-

ing pixels or in a pre-defined region. In particular for sky imagery, the LK method allows us to identify the dominant motion vector within cloud/sky regions, and therefore, is less sensitive to image noise. Wood-bradley et al [49] adopted the LK methodology to estimate cloud motion on images captured by a laptop camera. Instead of using the brightness intensity or the greyscale image, they converted the original image to the scale of blue-red difference to emphasize its prominent edges and corners. Subsequently, Wood-bradley et al extracted the prominent features at the points of interest with a large gradient of intensity for calculating optical flow and then manually removed the noisy features to refine the tracking result.

Many variants of the classic models of LK and HS were developed to improve the quality of optical flow. It is beyond the scope of this thesis to review the entire literature, and to search for appropriate techniques for the purpose of cloud tracking. Hence, we focus on several typical models that can effectively address the problem of cloud tracking. Chow et al [71] utilized a well-known *global* smoothing method in optical flow to detect the clouds' motions on the basis of a new sky imager system (USI [29]). They also adopted the robust estimation framework that originally appeared in [72]. Compared with their previous work on estimating the block-wise motion [46], the robust optical-flow approach, BA (short for Black and Anandan), is claimed to achieve more than a 19% improvement in cloud forecasting. Most optical flow models, including BA, adopted a coarse-to-fine warping scheme for constructing a multi-scale image pyramid and iteratively looping through the layers of images for stable outputs of optical flow. However in practice, the coarse-to-fine heuristic quite often does not perform well, particularly for the large movement of objects between two frames. The resulting optical flow fails to faithfully represent the real motion [73]. As fast-moving clouds are common in our observations, we therefore looked into the large-displacement optical flow (LDOF [73]) and compared it with other classic models.

2.3.2 Block-wise Motion Tracking Methods

A block-matching method tries to search for the best-matching blocks from two consecutive images that maximize the pre-defined criterion of similarity. Typically, the criterion of similarity in block-matching techniques is based on cross-correlation or least-square errors, and helps us to find the disparity vector between the same reference block across two images within a constrained window. Different from the OF methodology that assumes constancy in brightness at the pixel level, block-matching techniques assume that the reference block retains textural constancy over time.

Block-matching methods long have been in existence for cloud tracking in satellite imagery [65] [74] [67]. Most used cross-correlation to find the maximum correspondence of regions/blocks between two consecutive image frames [18]. In 2011, Chow el al. [46] adopted cross-correlation as the matching criterion to detect sparse cloud motion vectors (CMVs) in the images of TSI 440A, dividing the original TSI image into the blocks of identical size. As a TSI generates an image from the reflection of dome-shaped mirror, the regular blocks in a raw image are distorted and not uniform in real-space. Therefore, the image distortion compromises the accuracy of the detected motion vectors, especially around the boundary of an image. To resolve this issue, Huang et al. [21] [48] proposed preprocessing TSI images and transforming the original distorted view to a planar view. Thereafter, they located the best-matching blocks based on the Normalized Cross-Correlation (NCC) value, and utilized a refining threshold to remove the low accuracy matches, i.e., the low NCC value. To mitigate the deformation in cloud and predict the variation in block-wise motion, Huang et al. [48] investigated a multi-frame motion vector tracking and back-tracking and extrapolated the future movement with the detected motion trend. However, these approaches are not computationally efficient and can not take advantage of the cloud's information. To address these issues, we pro-

posed a new method, termed "cloud-block matching" (CBM), to dynamically determine cloud blocks in sky images [12], estimate the motion vectors only in the blocks of cloud pixels, and thereby assure its efficiency and efficacy via an intelligent clustering.

2.3.3 Miscellaneous Other Approaches

In contrast to those two types of tracking, other tracking approaches may use image registration techniques to attain disparity vectors, viz., object motion, between two consecutive frames. For example, phase correlation is a fast noise-resilient approach that estimates the translative offset between two similar frames or sub-frames in the frequency domain [75]. The phase-correlation method, as shown in the prior work [21], can not discriminate multiple movements within the same sub-frame region. More importantly, compared with the block-matching technique, the phase correlation technique incurs a higher error rate and is less accurate when no obvious cloud texture is available in the designated regions.

To better address the dynamics of cloud motion, and track the deformation over time, Bernecker et al., [50] [76] proposed using a non-rigid image registration. They used the well-studied diffusion model that is developed by Thirions Demons et al., [77], based on the optical flow method to detect a combined motion vector field with both global translative motions and local variations obtained from the diffusion model. Many new deformation models, such as the flow and curvature models [78], follow the Demons' method and are widely used in the medical image registration.

In addition, the Particle Image Velocimetry (PIV) methodology is used to estimate cloud motions as a velocity field. Chu [52] and Marquez [79] adopted the MPIV software that was developed by Mori and Chang [80] to detect the block-wise cloud velocity field in TSI images. Here, MPIV partitions an im-

age into reference blocks and searches for the best matching one based on the correspondence criterion of the Minimum Quadratic Difference (MDQ) or the cross-correlation coefficient within a nearby window. Afterward, MPIV applies the post-process steps of filtering and interpolation to smooth out the output velocity vectors. However, MPIV does not perform well when an image has discontinuities, such as the artifacts of shadowband and the camera's supporting arm in the TSI. To improve the accuracy and aggregate the sparse motion vectors of MPIV, Chu et al., [52] proposed using a k-means clustering to extract two majority motion clusters to differentiate stationary clouds and/or clear sky from fast-moving clouds.

In 2012, Huang et al., introduced a hybrid method that incorporates the stable local descriptor or local feature in estimating cloud motions [47]. On the top of the block-wise motion, Huang et al., used the Partially Intensity Invariant Feature Descriptor (PIIFD) [81] to adjust the motion vectors so to enhance the robustness to geometric and photometric variations. Similar to LK optical flow motion tracking, Huang's method can determine sparse motion vectors at those points with a large gradient of intensity and correct erroneous block-wise motions.

2.3.4 Block-matching v.s. Optical Flow

Block-wise motion estimation has the ability to utilize regional information, therefore is resistant to pixel-wise noise especially when the image is of low quality. Compared to variational methods, block-matching methods that are built upon the foundation of the cross-correlation correspondence or square errors within a reference window, are more robust to intra-frame brightness changes and color variations. If the underlying motions consist of only translation changes without any shearing and scaling, block-matching usually works really well.

However, accurate block-wise motion estimation requires a good segmentation in sky images to ensure that each reference block can faithfully represent an unique piece of cloud. In practice, as non-rigid clouds can have a variety of shapes and positions in image, the effectiveness of block segmentation is inconsistent. Especially for traditional block-matching methods or block-based techniques, e.g. BM and MPIV, blocks are predefined regardless of image color and textural feature. Consequently, the effectiveness of block segmentation varies with various distribution of cloud pixels and block sizes in image. Even with dynamic block allocation on the basis of cloud mask [11], the performance of locating the best match would be significantly compromised when multiple pieces of cloud reside in the same block.

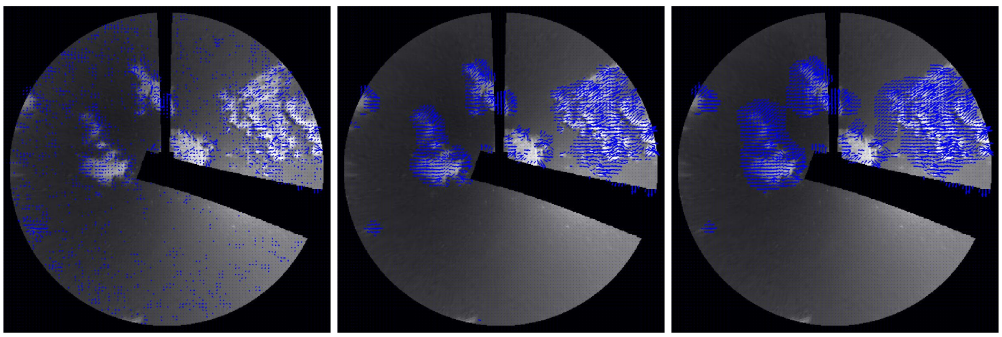
Another issue of block-wise motion tracking is that the performance is very sensitive to the block size. If the reference block is too large, it may contain several unrelated areas, such as stationary background which reduces the overall variation inside the block, and even generate undesired bias w.r.t. the cross-correlation score. When a block is too small, the local noise may dominate the best-match result. Therefore for different types of image sources and sky scenes, the block size must be carefully tuned [21] to acquire stable performance.

Moreover, the block-matching methods are inadequate to discriminate variations of motion inside the block unit. Particularly in sky imagery, block-matching methods cannot take the deformation or change of clouds over time into account. To compensate this drawback, prior works either divided images into very small blocks based on a hierarchical scheme [48] or applied interpolation to motion field based on overlapping blocks [80]. These approaches, however, acquired limited improvement and did not address the problem adequately.

Optical Flow (OF) methods address the motion tracking problem at a lower level than the block matching methods doe and try to extract pixel-wise motion through variational models that employ energy-like objective function, and

assume the consistency of spatial and temporal derivatives for their internal optimizations. Compared with block-wise approaches, the advantage of OF methods is the ability of accurately representing complex rigid motions such as rotation and scaling.

As the variational approach is sensitive to local noise, OF methods either add *global* regularization terms to smooth the dense motion field (as the HS model does) or estimate motion in a region/object (e.g. LK) to mitigate the noisy pixels and utilize structural information. However, their motion tracking results may be biased towards the input values acquired during the initialization in the variational settings. As described in [73], the solution often falls into a local minima that has the smallest motion because the majority of the OF approaches initialize the motion field as "zero".



(a) $\lambda = 1, iter_{\#} = 600$ (b) $\lambda = 11, iter_{\#} = 600$ (c) $\lambda = 16, iter_{\#} = 4600$

Figure 2-4: HS optical flow tracking with different parameter settings, λ: smooth factor, and $iter_{\#}$: number of iterations. Dense optical flows are denoted as blue arrows in the above images. (a) shows the example of outliers, (b) a large λ filters out noisy flows and velocities, and (c) however, a large λ potentially causes the problem of oversmoothing real cloud regions.

Another potential problem of applying the OF methods to detect cloud motions is the sensitivity to outliers and the associated parameter λ for *global* smoothness and outlier control. Figure 2-4 shows the results under different parameters when applying the OF methods (e.g. HS) to track the cloud motions in TSI images. The corner cases in this example, such as noisy pixels or artifacts, led to erroneous estimations of large velocity in Figure 2-4a. When

a large λ is applied to remove outliers in the iterative motion estimation, the optical flow may potentially oversmooth the areas around the cloud boundaries or corners that should have attained clear motion field segmentations. As we observed in Figure 2-4c, sky pixels near the cloud regions are falsely smoothed to acquire continuous motion vectors, as opposed to the ground truth of static motion field. As a result, when applying the OF methods for cloud tracking, we must carefully tune and customize the parameters to find the best-fit configuration for various scenes and sky images.

2.4 Solar Forecast Systems

2.4.1 Forecast Horizons

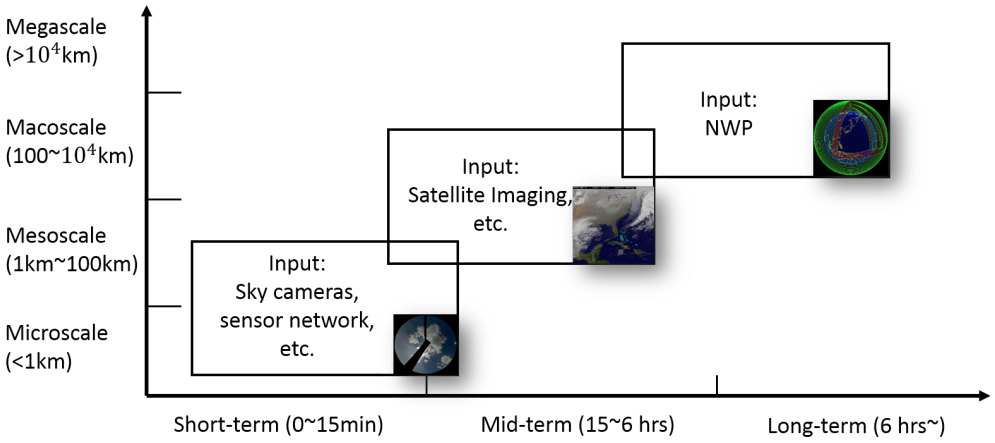


Figure 2-5: The categories of Solar forecasts and their input data sources.

We reuse the classification mechanism of atmospheric research to divide image-based solar forecasts into four spatial categories based on their targeted scale: microscale, mesoscale, macroscale and megascale, and three temporal classes for different forecasting horizons: short-, mid-, and long-term (See Figure 2-5). Each solar forecasting system has its own optimal temporal and spatial scales, consists of various core methods along its prediction pipeline that have

different input data and output forecasts.

Short-term forecast is often referred the prediction of solar variation from "seconds" to "minutes" ($\leq 15min$) ahead. It usually covers micro-grid of one kilometer or less, and ideally serves distributed solar panels and a small section of PV system. This type of approach often utilizes local observations or measurements, e.g. ground-based sky cameras or distributed radiation sensors as input and predicts the direct normal irradiance (DNI) or GHI as output in the aforementioned time range. With the help of localized cloud observations and irradiance changes, short-term forecast, in general, attains the best prediction accuracy in this time regime. The targeted applications of short-term forecast include, risk assessments of frequent ramp events during solar power generation, power quality assurance and control of micro grids containing renewable energy, minimization of spinning and battery reserves, and avoidance of frequent start-up and stop operations on spinning reserves. As operating schedules of PV system and solar panels get closer to real-time response, short-term forecast has been gaining significant interests from the participating members along the electricity value chain. On one hand, ISOs can evaluate, fine-tune dynamic operational schedules, and maximize the efficacy of solar power on the basis of accurate prediction of solar outputs given estimated shading/unshading section in their PV systems. One the other hand, short-term forecast is essential to optimize the schedule of associated storage systems or ramp-absorbing ultracapacitors [82]. It helps to mitigate the short-term irradiance ramps, as a result, enables grid operators to reduce the cost of counteracting voltage fluctuations via operating dynamic inverter or a secondary power source [83].

Mid-term forecast usually covers solar irradiance (e.g. GHI) between fifteen minutes and up to six hours ahead [84]. Mid-term forecast system has a large FOV and often deals with mesoscale or macroscale grids with the spatial resolutions ranging from several kilometers to up to hundreds of kilometers. Therefore, the appropriate input data for this type system is satellite imagery

generated from the radiometers in a satellite system that usually have a broad view as big as continent or the whole planet. The mid-term prediction is important for optimizing the operations of stand-alone solar systems and hybrid grids that contain high penetration of intermittent solar energy. The accurate forecasts of hours ahead are highly valued by utility companies, ISOs and solar power producers for core grid-wide operations and revenue-generating transaction agreements. Especially for hybrid grids, hours-ahead forecast of ramp events is crucial to load-following forecast, for instance, when cloudy condition is predicted to trigger a drop in solar production, grid operators increase the output from secondary power generation units to cater the load and stabilize production in the utility-scale.

Long-term solar forecast targets at the future production trend between hours or days ahead. Because most of clouds change significantly within this time regime and any solution solely on image-based observation loses its efficacy for such a long time window, we have to resort to the data-driven simulation model, a.k.a., NWP models, to generate low-resolution outputs of solar irradiance in the global scale ($\geq$ 100 km). Even with a coarse resolution, long-term solar forecast system plays a vital role as well in modern PV systems and hybrid grids. Firstly, the accurate hours- or days-ahead prediction is useful to estimate solar power production and more importantly, understand the long-term pattens of production load from various solar resources. Secondly, a robust long-term solar forecast system is useful for the assessment of unit commitment by utility-scale solar plant and subsequently supplies the plant with reliability evaluation to determine if and when the additional generation is needed. Therefore armed with this advanced knowledge, economic consideration can be integrated into the capacity expansion for obtaining the maximal profit by electric power industry. Thirdly, long-term forecast can also serve as the basis of the scheduling system for utility companies to optimize their transactions in electricity market. The accurate solar production load estimation is extremely useful for energy

service providers and grid operators to balance supply and demand of energy and to assure the reliable grid operation.

2.4.2 Literature Survey in Short-term Forecast System

Because of the increasing need of fast, proactive and smart grid operations in response to various events, short-term forecast gains attentions in modern grids due to its proven ability to improve PV system's power output with the knowledge of potential shading/unshading events in advance. Based on the categories of solar irradiance forecast in [85], the short-term forecast often relies on sensor network or ground-based sky imagers to extract temporal information and apply stochastic/machine learning methods to learn from extracted input information, and regress to output irradiance.

Sensor network consists of solar irradiance measurement sensors or output meters that are strategically deployed to cover particular areas and locations. It is useful and cost-efficient to forecast potential rise/drop of solar irradiance within the sub-minute range and helps better control ramp-absorbing ultracapacitors [82]. The core idea is to derive cloud information, such as velocity and direction, based on spatio-temporal analysis of available sensor data. In particular, this type of approach first collects series of readings from well arranged radiation sensors and meters, performs temporal correlation and causality analysis between any pair of spatially distributed sensors, then associates the same irradiance ramp event with a sequence of sensors, and ultimately calculates the actual cloud velocity and direction based on the learned time span for a ramp event traveling across a sequence of sensors whose locations are known as prior [9] [86] [87]. Limited by the scale of fixed radiation sensors, this type of approach usually requires that distributed sensors be close to enough. Without validation by the real overview of sky, its performance may be compromised when the measurements from radiation sensors are not highly correlated in a

certain time window due to complex cloud conditions.

The majority of recent research efforts utilize images collected from ground-based sky cameras. Sky imagery was proven to be effective in predicting the presence of clouds based on the temporal correlation between frames. The core idea of this approach is predicting the cloud properties in and around the path between the radiation sensor and the Sun based on previously observed clouds and then estimating the cloud-induced fluctuations in irradiance based on these predicted properties [46]. By representing irradiance as a function of the features extracted from sky images, solar prediction can then be transformed into the problem of predicting future images using persistent motion vectors [28] [11] or by solving the advectiondiffusion equation [88].

Most skycam-based forecasts cover the time range from half minute to up to 15 minutes due to the limitation of FOV and image resolution. A recently study shows that the forecast error of one sky camera (i.e. TSI) ranges from 18 to 24% for different forecast horizons [30]. Within such a short time window, the persistent model [89] performs extremely well, and therefore any new solar forecast needs to minimize errors in motion tracking and irradiance modelling in order to deliver a better performance than the persistent model.

To better learn the uncertainty of detected cloud motions from sky images, Huang et al., [48] introduced statistical features to the final irradiance forecast model. Being aware of the instability of motion tracking, one recent effort evaluated the uncertainty of motion vectors in designated areas, and further assimilated the uncertainty values as a new type of feature into a stochastic model, and as a result, improved irradiance forecast [90]. To further improve the accuracy of short-term solar forecast, recent works have incorporated various machine learning techniques into both cloud tracking and irradiance modeling [52] [53]. Considering the drawbacks of single camera device, we proposed to integrate multiple image streams from several spatially distributed cameras to reduce the observation error and track clouds with spatio-temporal correlations [11]. Be-

cause these new solar forecast systems are image-based, the quality of image feature extraction has significantly impact to the intermediate tracking results and final outputs. In details, they use obtained motion vectors to generate predicted cloud "images", extract multiple features for these "images", and input them to a chosen regression model for the final output [91].

2.4.3 Literature Survey in Mid-term Forecast System

Nearly all mid-term forecast system rely on satellite images as input to predict the variations of solar irradiance. Though a plethora of research efforts investigated satellite-based models to derive accurate surface irradiance using statistical or physical methods [92] [93], their results, nevertheless, can not be directly used for solar forecast. To integrate the satellite model into forecast, an image-based cloud tracking scheme is also necessary for predicting cloud information in advance. Previous approaches applied block-wise cross-correlation for matching clouds and generating cloud motion vectors [65] [94]. These matching technique often assumes invariant cloud texture and constant cloud movements [95] [96]. However, this assumption is often violated in reality, which renders the most challenging problem in these studies. In fact, cloud condition spanning several hours apart is often volatile because of cloud formation, movement, deformation and disappearance. The situation is further complicated by the unstable performance of motion tracking algorithms when they are applied to satellite images. Instead of using direct motion tracking algorithm, many recent works tried to regress multiple satellite frames of previous time stamps to the frame in the future, and then integrated other sources of information, e.g., ground radar measurement, into prediction. We term this type of approach "no-motion method". Here the ground radar based prediction model adopts the continuous data stream of radiation fluctuation pattern to improve its performance [97]. But those no-motion methods tend to generate

low-precision outcomes and offer only forecasts with a short time horizon. Because real cloud tracking is not involved, the forecasting model is essentially stochastic, and therefore has a limited prediction range between several minutes and up to two hours [98].

Satellite models were originally designed for correlating cloud coverage with Global Horizontal Irradiance (GHI) [93] [99]. Lately, this original idea still appeared in some recent works, for example, two separate efforts studied the linear relationship between Direct Normal Irradiance (DNI) and satellite visible channel [100] [61]. These models used the term "Cloud Index" (CI) to represent the optical density derived from satellite data [101]. CI is widely used for measuring solar absorption by clouds, and determined by the cloud fraction or coverage at a particular ground location as observed by satellite. By using multiple empirical clear sky models, the local solar energy distribution is derived from satellite images [102] [103] [104] [105]. Some recent work refined this model by adding extra parameters/factors, such as terrain factor [96]. To improve the accuracy of cloud index in satellite image for various cloud types and under different lightning conditions, several research efforts proposed to integrate multi-spectral data to attain robust cloud identification, classification and snow detection [106] [107].

A large number of previous works in mid-term forecast investigated new techniques to improve the model performance and identify complex cloud movements in satellite images [108] [109] [10]. In contrast to the cloud-motion based approach, several recent studies on solar forecasting proposed some new ideas of exploring the established techniques of other fields, such as statistical approach [110] and Artificial Neural Network (ANN) [111]. In particular, some state-of-the-art models does not even require meteorological data [112] and only relies on analyzing time series of irradiance to derive predicted values [113]. A recent work shows that the vector autoregressive (VAR) model on ground-based observations significantly lowered the root mean square error of forecast

by 35% [114].

Although satellite images were applied to days-ahead solar irradiance forecast [60], given the orbit property of a satellites, the methods based on satellite images usually are optimal for mid-term forecasts that have the temporal resolution between half hour to six hours and the spatial resolution of several kilometers [59]. The forecast model of many satellite approaches is essentially a linear regression between detected cloud coverage and solar irradiance. However, it turns out that the simple linear relation derived from cloud coverage is not capable to capture and describe variation under cloudy condition. Therefore, a customized constant must be used as an estimation compensation to reduce linear biased influence [64]. Furthermore, satellite image models can be combined with other types of forecast models in a hybrid system. As a result, the hybrid approach is adaptive over various forecast horizons while attaining reasonable performance over its individual component models [79] [7].

2.4.4 Literature Survey in Long-term Forecast System

In general, long-term forecasts rely on macroscale model and assimilate low-resolution data for improving model. They usually apply various classic numerical methods to a large grid, and foresees the availability of solar power in a continental or global scale. The method is often built on top of a big data analysis engine that integrates various types of measurements. Long-term solar forecast based on numerical prediction model (NWP) requires knowledge about the initial state of the global atmosphere and the physical laws which govern the evolution of the atmosphere [85]. The solar radiation of each grid cell is already one of many numerical outputs of NWP model [115]. Current solar forecast systems utilize various well-known NWP models, including Global Forecast System (GFS) and Weather Research and Forecasting (WRF) model [116] [117].

Chapter 3

Cloud Detection and Tracking in Sky Imagery

Cloud is the primary factor to cause solar irradiance fluctuations and becomes the focus of many solar forecast systems. In this chapter, we discuss a new method to identify cloud pixels and tracking cloud movement from a stream of images and then evaluate its efficacy using different types of images taken by various sky cameras. We detail our pipeline for assimilating cloud information and tracking complex cloud motions. Firstly, we summarize traditional threshold-based approaches and analyze their weaknesses, and then propose to apply a supervised classifier to determine clouds from images at the pixel level. Thereafter, we aggregate detected cloud pixels into regions/blocks for estimating block-wise motions and extracting spatial information. Because motion tracking plays a vital role for forecast quality, we thoroughly investigate several state-of-the-art motion tracking models that are already adopted or have great potential for detecting and tracking cloud motions. Based on the analysis of classic models and techniques, we design a hybrid model to obtain the advantages of block matching and optical flow. Finally, we evaluate the new hybrid model in different types of images, including synthetic image sequences and real image streams from multiple HD sky imager deployed at various sites.

3.1 Cloud Detection

3.1.1 Cloud Pixel Identification

To extract cloud information in image, pixels of cloud need to be identified and differentiated from others e.g. clear sky pixels. However as clouds may have distinct color and patterns in image, it has been a challenging problem in image-based research. Moreover, under different weather conditions and at different solar angles, the clouds presented on a bitmap image may appear to have various brightness and a large range of intensities. Therefore, we require a robust methodology to assess sophisticated image-based features to capture the pixel-wise differences or the regional textural differences between cloudy and clear regions. From the perspective of image processing, it may appear that we could first sharpen edges and boundaries using various derivative image filters and then apply image segmentation methods to separate clouds from clear pixels based on their boundary pixels, and thereby identifying cloud segments. However, these methods often fail to segment clouds from the background due to the poorly-defined edges of clouds in sky images. With the resolution and image quality limitations, a single pixel in a sky image may contain both cloud and clear sky. Consequently, a section of visually rigid cloud can appear non-rigid and blurry in sky imagery.

Furthermore, cloud detection in sky imagery can be difficult as sky imagers face certain instrument-specific challenges in their practical deployment in the field and in testing. Because the sky camera adjusts its lens aperture and shutter speed in response to the amount of incident illumination, the output images can potentially suffer from variations in exposure, and may appear either brighter or darker than the ground-truth image that accurately represents the real lighting conditions.

To accurately detect cloud pixels, previous studies of sky images have relied on cloud properties and have utilized the prominent features at the pixel level to

identify hybrid thresholds for the RGB channels or the red-blue ratio (RBR) in cloud images [13] [35]. The threshold-based method presented in [35] is able to accurately identify opaque clouds in images acquired by the Whole Sky Imager (WSI). However, it is less effective for many low-quality commodity cameras, such as TSI and webcam. These cameras lack spectral and neutral filters and adjust their lens apertures for different lighting conditions, which incurs large variations in the brightness of the images and requires non-trivial calibrations in deployment. Under this circumstances, threshold-based methods require customized parameter settings for various sky scenes. Another type of approach to the analysis of cloud properties is based on supervised classification methods that utilize various features extracted from sky images and find patterns from the training dataset. In earlier works, traditional classification techniques, such as k-nearest neighbor (kNN) algorithms, binary decision tree, and neural network classifiers [39] [40] [42] [118], have been widely used for cloud type classification in sky imagery. To detect clouds at the pixel level, a supervised technique can be successfully applied to train the adaptive threshold for the classification of image pixels into several opacity categories based on the RBR channel [36]. In this paper, we detail how to use a binary classifier to identify cloud versus clear-sky pixels in sky imagery. The training datasets for the classifier are generated by manually labeling cloud/sky pixels in TSI images. This process requires a considerable amount of human effort and more importantly, may introduce uncertainties and errors into the training datasets that could significantly impact the accuracy of traditional classifiers.

3.1.2 Cloud Detection Pipeline

To overcome the aforementioned challenges, we propose an outlier-aware classifier to train manually labeled pixels in sky images. Moreover, because of the many instrument-specific abnormal cases that can be generated from individ-

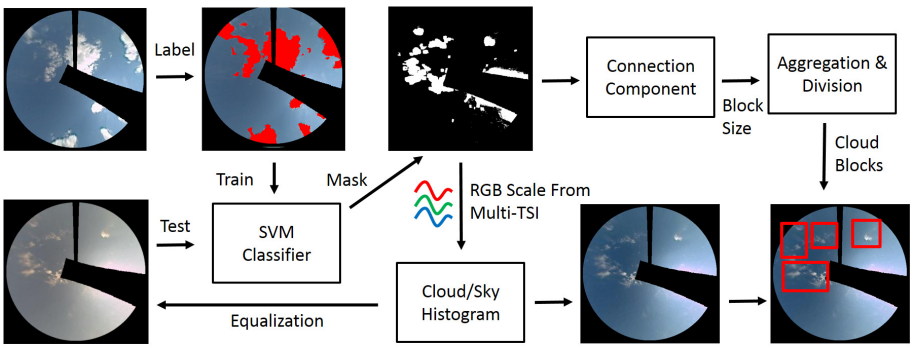


Figure 3-1: Pipeline for cloud detection using an SVM classifier and multi-source correction.

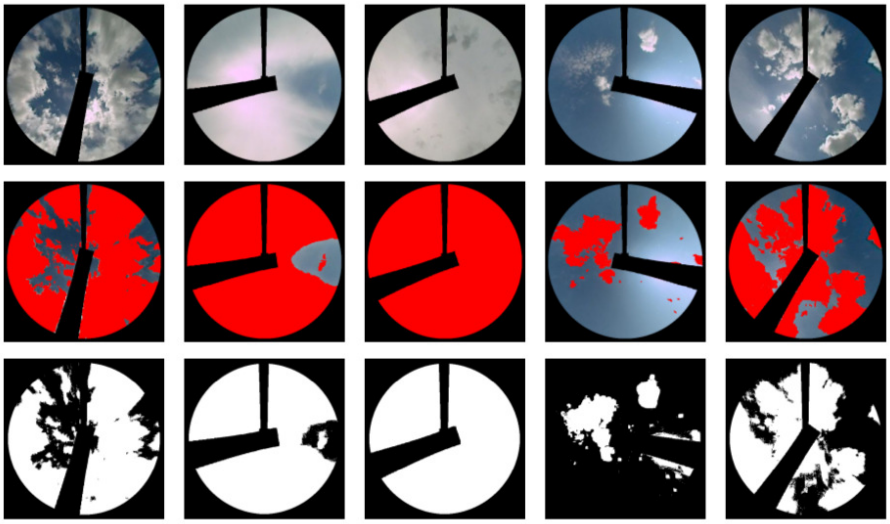
ual TSIs, such as, overexposed or underexposed images, we design a classifier-based pipeline to utilize all three TSIs for multi-source image correction to enhance the overall accuracy of cloud detection. We also present a technique for aggregating cloud pixels into cloud blocks (Figure 3-1).

To minimize the influence of possible outliers in the training dataset, we chose a Support Vector Machine (SVM) [119] as a pixel-wise classifier and trained it using two-layer cross-validation [120] to reduce overfitting. An SVM constructs a max-margin hyperplane to reduce the effect caused by outliers and offers the advantage of being able to handle known outlier patterns. In particular, uncertainties and errors introduced during the manual annotation of training images are taken into account during SVM optimization. Figure 3-1 shows an example of the training and testing process for a cloud classifier. To better describe the characteristics of cloud pixels, six features are extracted from the sky images and normalized for use as the SVM training dataset: R, G, B and RBR which are spectral characteristics extracted from the RGB color space, and two features based on the neighboring pixels which are used to mitigate the impact of variations in illumination. In detail, we convert the sky images from the original RGB color space to the relative luminance space based on the algorithm presented in [121] and then apply the Laplacian of Gaussian(LoG) spatial filter [122] to compute the 2nd derivative in luminance space within a

fixed-size window (7x7 is used). The LoG value reflects the rapid changes in the illuminance channel in a certain region and is useful for detecting sharp edges. The final feature used for cloud detection is the standard deviation of the luminance channel within a small region around each pixel (7x7 is used), representing the pixel's average difference from its neighbors. Our implementation uses the SVM package *libsvm* [123] with linear kernel.

Furthermore, we explore the possibility of multi-source abnormality correction using synchronized images from all three TSIs. Because the three TSIs are located reasonably close to each other, we can reasonably assume that their color representations have statistically similar range with regard to the RGB channels. In other words, cloudy/clear pixels in these three digitized color channels should have similar histograms across the different TSIs. Therefore, if one TSI experiences an exposure issue or abnormal brightness in the RGB color space, we can correct it by equalizing the histograms of its RGB channels to those of the two normal TSIs. We first generate the cloud masks for all TSIs using the SVM classifier and compute the histograms of the RGB channels for cloud and sky pixels separately. By calculating the Euclidean distances between the histogram vectors of the three TSIs, we can identify a device as abnormal if its image histogram is significantly different from those of the other two devices. We then apply the histogram equalization add-on to the output image from the abnormal TSI to adjust its RGB scale for cloud and sky pixels. The corrected result can be used for the next round of cloud mask generation. In practice, we iterate this procedure three times to extract the cloud mask and equalize the RGB histograms of the TSI images (Figure 3-1).

To evaluate the performance of the proposed supervised classifier and the multi-source correction algorithm, we selected various test cases from daily observations corresponding to different atmospheric conditions and cloud types and then compared the results with our manually annotated images. In this paper, we adopt two evaluation metrics to measure the error in cloud classification:



$AC_{cld} = 93.2\%$	$AC_{cld} = 100\%$	$AC_{cld} = 100\%$	$AC_{cld} = 83.2\%$	$AC_{cld} = 93.8\%$
$AC_{sky} = 87.7\%$	$AC_{sky} = 79.4\%$	$AC_{sky} = 0\%$	$AC_{sky} = 92.9\%$	$AC_{sky} = 85.5\%$

Figure 3-2: Cloud detection results (row 3) compared to manual annotation (row 2) under different weather/cloud conditions. *Left to Right* in row 1: Scattered cloud, cloudy, overcast, multi-layer, and multi-layer with thin cloud. AC_{cld} and AC_{sky} represent the accuracy of the classification results for cloud pixels and sky pixels respectively.

$AC_{cld} = 97.9\%$	$AC_{cld} = 98.0\%$	$AC_{cld} = 86.1\%$	$AC_{cld} = 98.9\%$	$AC_{cld} = 96.2\%$
$AC_{sky} = 91.5\%$	$AC_{sky} = 84.8\%$	$AC_{sky} = 91.9\%$	$AC_{sky} = 81.9\%$	$AC_{sky} = 91.0\%$

Figure 3-3: Results of cloud detection (row 3) compared to manual annotation (row 2) in the presence of device-specific bias or luminance variations. *Left to Right* in row 1: red-dominant, green-dominant, overexposed, underexposed, and a different image source.

$$AC_{cld} = \frac{N_{cld,cld}}{N_{cld,cld} + N_{sky,cld}}, AC_{sky} = \frac{N_{sky,sky}}{N_{sky,sky} + N_{cld,sky}} \tag{3.1}$$

where AC_{cld} and AC_{sky} are the accuracy of cloud and sky pixel classification, respectively. $N_{cld,cld}$ and $N_{sky,sky}$ denote the pixel counts of correct cloud and sky classifications, respectively, whereas $N_{sky,cld}$ and $N_{cld,sky}$ indicate the total numbers of sky and cloud pixels, respectively, that are falsely recognized by our detector. In Figure 3-2, we show the distinctive patterns of examples of scattered clouds, cloudy conditions, overcast conditions, and two multi-layer cases. Compared with the manual classification masks, our pipeline based on the SVM classifier can accurately detect clouds (with an accuracy of more than 83.2%), except in the case of multi-layered clouds near the sun's position and very thin clouds. The image area near the Sun ("sunspot") has a higher brightness and is difficult to characterize based exclusively on static textural information. Therefore, the classifier often falsely labels clear-sky pixels as clouds in that region.

Table 3.1: Overall confusion matrix for the cloud detection pipeline applied to the images shown in Figures 3-2 and 3-3. *Left to Right in the Table*: AC_{cld} followed $1-AC_{cld}$ in the first row and $1-AC_{sky}$, followed by AC_{sky} in the second row. All values are in [%].

	Cloud Detection Pipeline	
Manual	Cloud	Sky
Cloud	96.6%	3.4%
Sky	10.3%	89.7%

To validate the cloud detection performance of our method in the case of device errors or variations in exposure, we selected four abnormal images of types that are commonly observed in the field: red-color dominant, green-color dominant, overexposed, and underexposed (the first four cases in Figure 3-3). The accuracy of cloud pixel classification in these cases is 86.1% or higher. We also applied our classifier to another type of sky imagery configured with a different field of view and color scales (the last case in Figure 3-3), which confirmed that our classification algorithm is practical and effective. The classification accuracies for cloud and sky pixels in this case are 96.2% and 91.0%, respectively. The overall clout detection performance in all selected cases is evaluated in the confusion matrix presented in Table 3.1. We observed our pipeline accurately (96.6%) recognizes cloud pixels.

3.2 Cloud Block Aggregation

The cloud pixel distribution detected by supervised classifier is often utilized to aggregate into regions/blocks for motion vector discovery [11]. The core idea is to divide the cloud mask into individual regions. If the texture inside is not "good" w.r.t. similarity criterion defined in motion tracking, the region is often divided or merged with its neighbors to include more information [12]. It significantly helps to improve the effectiveness of the best-match finding in block-wise motion detection and as well as reduce the complexity of computation by

removing non-cloud pixels during calculation.

3.3 Classic Motion Estimation Models

In this section, we investigate seven classic models that effectively track cloud motions in sky images, detail their implementation, and explore the appropriate parameter settings guided by state-of-the-art works.

3.3.1 HS Optical Flow

The original **HS** formulation, proposed in [69], enforces two constraints: The intensity conservation, and the *global* smoothness among optical flows, and minimize the following energy function:

$$E(u,v) = \sum_{x,y}(u \cdot I_x + v \cdot I_y + I_t)^2 + \lambda(\|\nabla u\|^2 + \|\nabla v\|^2) \tag{3.2}$$

where u and v are the movement in the x and y direction respectively at each position (x,y). ∇u and ∇v are the gradients of u and v, and can be approximated by subtracting the value at the point (x,y) from a weighted average of the values at its neighboring pixels [69]. The iterative algorithm of **HS** uses the regularization weight, λ, to control global smoothness. To achieve optimal performance, we need to tune the smoothness parameter λ carefully for different applications. Even though the heuristic solution of **HS** long has been deemed inaccurate, it still achieves surprisingly good results with a proper optimization and parameter tuning [124].

3.3.2 LK Optical Flow

The **LK** methodology [70] relies only on the local information from a small window surrounding the point of interest. Essentially, it is a *local* approach, and

can faithfully extract the motion vectors at the points that have prominent features, such as corner points and edges [49]. The energy-like objective function is defined as follows:

$$E(u,v) = \sum_{x,y} g(x,y)(u \cdot I_x + v \cdot I_y + I_t)^2 \tag{3.3}$$

where $g(x,y)$ is the weight function to determine the region wherein optical flow is constrained. The Gaussian function is a common example of $g(x,y)$ [125]. Eq. 3.3 assumes that all pixels within the region share an identical movement. We used the least-square estimator to resolve the optimization of the objective function. During our experiment, we used the *Matlab* implementation of **LK** that entails a coarse-to-fine framework [126]. We adopted the same settings following [49], converted the original sky images to the scale of blue-red difference, and then used a three-level image pyramid to iteratively extract **LK** ($iter_{\#} = 3$). $g(x,y)$ serves as a mask function and represents a region of 15 x 15 pixels. However, in contrast to the original work in [49], our approach generates a dense flow field for all pixels, instead of only for those positions with prominent features.

3.3.3 BA Optical Flow

In 1996, Black and Anandan introduced non-convex robust penalty functions into the objective of optical flow to replace the original square-error term [72]. Their method, denoted as **BA**, has been widely applied for various purposes, such as tracking objects and estimating motions in the field of computer vision [127]. In contrast to previous optical flow approaches, **BA** formulates the energy terms with a series of non-quadratic robust penalty functions $\Psi(x^2)$ for both the data term, E_{color}, and the regularization term for smoothness, E_{smooth}:

$$
\begin{aligned}
E(u,v) &= E_{color}(u,v) + \lambda E_{smooth}(u,v) \\
&= \sum_{x,y} \Psi_D((u \cdot I_x + v \cdot I_y + I_t)^2) + \lambda \cdot \Psi_S(\|\nabla u\|^2 + \|\nabla v\|^2)
\end{aligned}
\tag{3.4}
$$

where Ψ_D and Ψ_S respectively are the selected robust functions for data and the smoothness penalty. Ψ_D and Ψ_S often are chosen to be the same, e.g. the Lorentzian penalty $\Psi(x^2) = log(1 + \frac{x^2}{2\sigma^2})$ in [128], or the quadratic penalty term $\Psi(x^2) = x^2$ in the **HS** method. Because Chow et al. [71] already demonstrated a good performance in tracking cloud motion by utilizing Eq. 3.4, we evaluate a well-known implementation of **BA** method developed by Sun et al. [127] and set $\lambda = 0.1$ based on the experiments in [71]. We also choose the Lorentzian penalty with $\sigma = 0.03$ as the robust function Ψ_D and Ψ_S. In contrast to the original work of **BA**, this new implementation explores the most recent techniques, including preprocessing, filtering, and interpolating methods. Consequently, it significantly improves the accuracy of motion estimation and eliminates noise in the flow field.

3.3.4 Large Displacement Optical Flow (LDOF)

Existing optical flow methods often have difficulties in handling small but fast-moving objects. To overcome this shortcoming, Brox and Malik introduced a new model to track large displacements [73]. Their method, denoted as **LDOF**, formulates the tracking problem with additional energy terms, and, more importantly, incorporates one extra descriptor that regularizes the objective. The idea is similar to the block-wise pipeline designed by Huang et al. [47] and Wood-Bradley's **LK** model [49], each of which generates sparse motion vectors based on *local* or prominent feature matching. Compared with the pixel-to-pixel match used only in those aforementioned approaches, descriptor matching relies on rich local descriptors, such as SIFT and HOG, and is useful for tracking the motions of structures and objects. Hence, this technique was applied suc-

cessfully to estimate arbitrarily large displacements between frames [73] or efficiently correct erroneous motion vectors [47]. The objective function of **LDOF** that is used to estimate pixel-wise motions consists of the following five energy terms:

$$\begin{aligned} E(u,v) = E_{color}(u,v) + \lambda E_{smooth}(u,v) + \gamma E_{grad}(u,v) \\ + \beta E_{match}(u,v,u_1,v_1) + E_{desc}(u_1,v_1) \end{aligned} \tag{3.5}$$

Note that different from E_{color} term in Eq. 3.4, which adopts a linearized form and thereby favors the estimation of motion field with small displacements, E_{color} in Eq. 3.5 involves no linearization so to retain the capability of tracking a large displacement of (u,v). This is important in estimating cloud motions because fast cloud movements that appear as large displacements between frames are common in sky imagery. E_{color} between the first frame I_1 and next one, I_2 is defined as follows:

$$E_{color}(u,v) = \sum_{x,y} \Psi(\|I_2(x+u,y+v) - I_1(x,y)\|^2) \tag{3.6}$$

E_{smooth} is identical to the second penalty term in Eq. 3.4 while $E_{grad}(u,v)$ is the supplementary constraint that assumes the constancy of intensity gradient at corresponding points on two frames:

$$E_{grad}(u,v) = \sum_{x,y} \Psi(\|\nabla I_2(x+u,y+v) - \nabla I_1(x,y)\|^2) \tag{3.7}$$

The remaining two terms, E_{match} and E_{desc}, are the energy terms related to the sparse descriptor matching. γ, λ and β are the parameters that need to be manually tuned based on the various sky images. To improve the performance of motion tracking, Brox and Malik investigated three different methods of descriptor matching: SIFT, histograms of oriented gradients (HOG) and geometric blur (GB). The results show that the HOG method generates the smallest number of mismatches in various dataset. Following their experiments, we applied

LDOF with HOG descriptor matching and a Gaussian filter with $\sigma = 0.8$ for cloud tracking in sky images. Moreover, we optimized the parameter setting of **LDOF** based on the model's performance on a training dataset of sky images. In summary, the combined parameter setting consists of $\lambda = 30$, $\beta = 300$, $\gamma = 5$.

3.3.5 Hierarchical Block Matching (BM)

In 2011, Huang et al.(**BM** [21]) proposed a two-layer hierarchy of image blocks on greyscale TSI images to guarantee an appropriate granularity for meaningful block matching, i.e., containing sufficient information for the matching calculation, whilst capturing detailed cloud movements. Accordingly, Huang et al. first preprocessed TSI images, divided them into blocks, denoted as reference or inner blocks that, in general, are small enough to contain only uniform movements. However, small (inner) blocks usually do not have adequate variation in image intensity, so that any block-matching for them may yield inaccurate results. Thereby, instead of using reference block directly, Huang et al. employed a bigger block (outer block) co-centered with each inner block to search for the best match across images. In addition, to refine the matching results and mitigate the influence of artifacts on TSI image, they undertook post-processing to the matching results and applied two thresholds to discard the blocks that have only all black/invalid pixels and/or low matching score. In this paper, we adopted the **BM** method with the same parameter settings outlined in [21].

3.3.6 Cloud-Block Matching (CBM)

To incorporate actual cloud information into block matching, we designed the **CBM** method outlined in our prior work [12] to dynamically determine and match actual cloud blocks. We applied the Support Vector Machine (SVM [119]) to generate a cloud classifier to separate cloud and sky pixels on the basis of manually annotated images. It utilizes multiple image features to categorize

image pixels into two classes (cloud/sky). Once the cloud mask is extracted from a TSI image, a connected component detection algorithm [129] is applied to the mask to generate the cloud regions that are separated from sky pixels. We assumed that each individual region is a piece of cloud with uniform motion inside, and drew a minimum bonding rectangle for each region to represent the initial cloud block. Lastly, we applied a split-and-merge scheme to divide large blocks or merge small ones, so to ensure that the output of cloud blocks is suitable for the subsequent block-matching computation. Similar to **BM**, the matching criterion of **CBM** is also based the NCC score. In the actual implementation, we specified the range of output block size between 10 x 10 to 80 x 80. The initial blocks that are outside the size range will be divided or merged.

3.3.7 Particle Image Velocimetry in Matlab (MPIV)

MPIV is a Particle Image Velocimetry (PIV) software package to analyze consecutive frames and obtain velocity vectors [80]. It generates motion vectors via a hierarchical search, and more importantly, post-processes the generated motion vectors with various de-noising techniques, such as a median filter, an iterative search, and/or motion interpolations. As a result, **MPIV** is widely adopted for detecting motions in various types of images, in particular, tracking clouds successfully in the images from satellites, or ground-based sky cameras [52, 79, 130]. In this paper, we used it to generate velocity fields of clouds across various sky imagers, and compare the results with other state-of-the-art models. Based on the parameter settings in [79] to obtain cloud motions in the TSI images, we chose a 32 x 32 pixels processing window and set the overlapping ratio of the search window as 0.5. In addition, we selected the Minimum Quadratic Difference (MDQ) as the matching criterion and set the recursion mode as two to iterate the entire process twice so to enhance the accuracy of the motion vectors.

3.4 New Hybrid Model

To address the aforementioned issues of optical flow and block-matching, we propose a new hybrid approach that integrates the block-matching method and the variational OF model, and uses the former method to guide/refine the latter one. This new model encompasses three main steps: (1) Extracting cloud mask, generating cloud blocks via bottom-up merging and detecting block-wise motions, (2) Identifying dominant motion patterns from detected motion vectors, and, (3) Estimating optical flow using our new formulation and refining based on multiple motion filters.

3.4.1 Cloud Block Generation and Matching

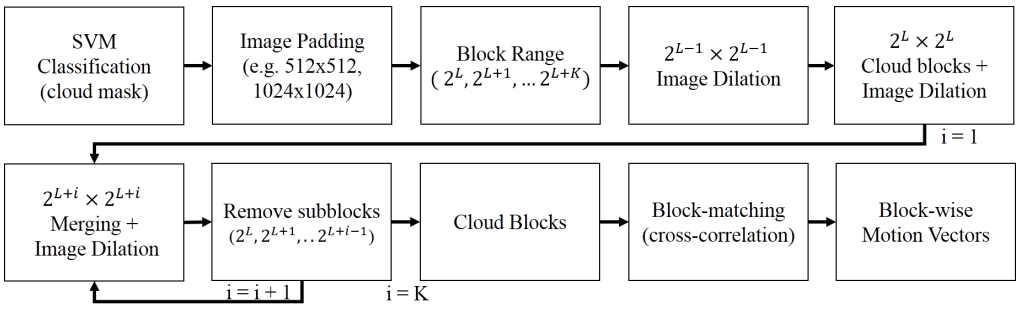


Figure 3-4: Pipeline of QCBM.

Because the block's size and position play a vital role in tracking clouds, we devise a more effective algorithm for block generation and matching than prior approaches, i.e., the Quadtree-based Cloud Block-matching (QCBM), so to take into consideration the cloud distribution in sky images. Here, a Quadtree recursively decomposes images into four equal-sized square sub-images until the criterion of homogeneity is met, or the minimum block size is reached, and consequently represents an image in a top-down hierarchy with different resolutions [131]. Such a representation can efficiently divide an image into 2-D *homogeneous* (similar color, texture, or structure) regions/blocks, and ease the

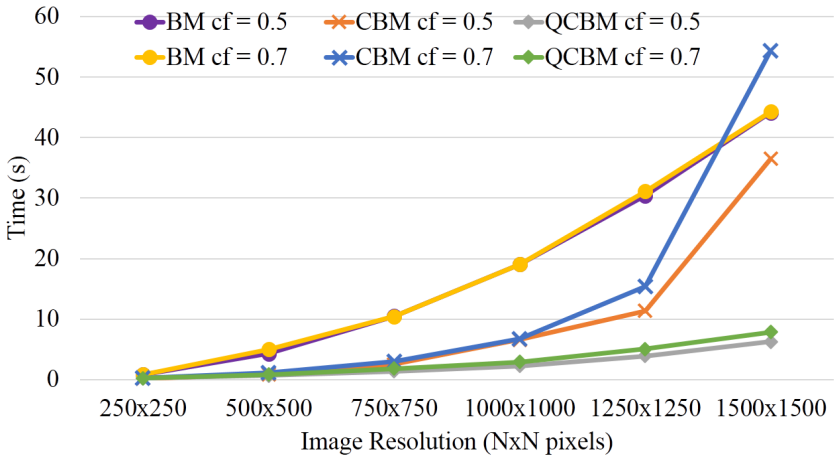


Figure 3-5: Execution time of block-matching algorithms for different image resolutions. *CF* indicates the cloud fraction of a sky image.

subsequent tracking process.

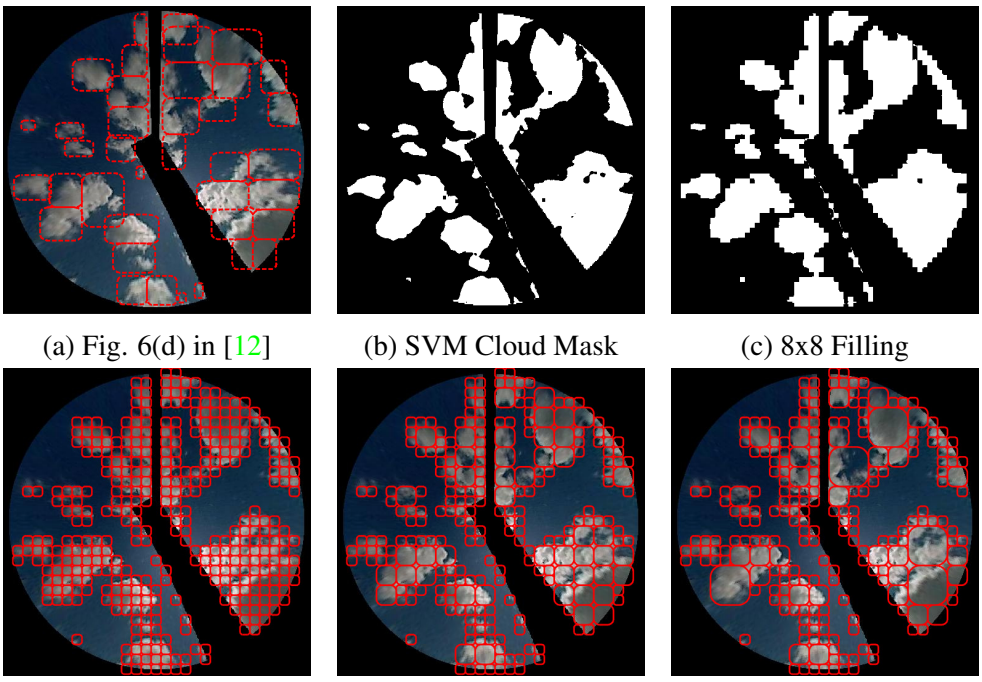
However, regions of low homogeneity, e.g., edges of clouds in the image, require excessive decompositions to meet the stopping criterion. As a result, a top-down decomposition usually generates cloud blocks that are too small to preserve meaningful textural information for motion tracking. To overcome this, we introduce a bottom-up scheme to the Quadtree construction that starts from all nodes at a pre-defined level in the Quadtree, identifies all cloud blocks containing cloud pixels more than the threshold *CF* (the fraction of cloud pixels in a block based on the cloud mask obtained in prior steps), selectively fills all pixels as clouds in these nodes, and iteratively merges homogeneous blocks.

Figure 3-4 shows the steps of the QCBM generation and matching. For simplicity in implementation, we pad the sky image to be square with the size of the power of 2. To ensure the quality of block-matching, we limit the dimension of the generated cloud blocks to be in a range of $(2^L \times 2^L, 2^{L+1} \times 2^{L+1}, ..., 2^{L+K} \times 2^{L+K})$, where 2^L is the minimum size allowed in generating cloud blocks. Under this new approach, there are only $K+1$ different sizes of cloud blocks. To ensure the quality of segmentation near the cloud's boundary, we first start at one layer lower than the finest level, i.e., 2^{L-1}, to fill the cloud mask if the cloud

fraction inside is beyond CF. During the iterative merging step from the level of 2^L in our bottom-up scheme, we check the four children/subblocks of each node: if no less than T children are cloud blocks, all four child nodes are eliminated, and the parent node becomes a new (bigger) cloud block, which will be filled with cloud pixels and participates subsequent merging, until the level 2^{L+K} is reached. Lastly, the Quadtree contains different sizes of cloud blocks for matching, as shown in Figure 3-6f. We then find the best match for each cloud block to find its block-wise movement to the next frame.

To obtain the optimal matching quality based on the correlation criterion, we choose three layers starting from the minimum block size 16×16 for sky imagery ($L = 4$, $K = 2$), and then we set $CF = 0.2$, $T = 3$ to fill and merge the lower level blocks. Figure 3-6 presents the result of the original block generation by the QCBM, based on the example in [12]. Figure 3-6c shows the cloud blocks with the starting size of 8×8 filled with the original cloud mask in Figure 3-6b, whereas Figures 3-6d- 3-6f present the iterative merging results of cloud blocks with a size from 16×16 to 64×64. We observe that compared with the blocks generated by **CBM** (Figure 3-6a), the blocks by the bottom-up scheme in QCBM (Figure 3-6f) have cleaner segmentation around the boundary of clouds and do not contain overlapping regions.

More importantly, this QCBM is computationally more efficient. We implemented QCBM, **BM**, and **CBM** in *Matlab* and ran them on a laptop. Figure 3-5 shows the average running time of three different block-matching algorithms under different image resolutions and cloud fractions. As the QCBM does not involve the "dividing and conquer" scheme in **CBM**, and eliminates most clear sky pixels during the block generation in contrast to **BM**, its execution speed is significantly faster than the others. Especially when an image is large with high resolution, QCBM only incurs a marginal increase in the execution time, and always completes the matching process within 10 seconds. Though the QCBM segments an image into more cloud regions when the fraction of cloud in the



(a) Fig. 6(d) in [12] (b) SVM Cloud Mask (c) 8x8 Filling

(d) 16x16 cloud blocks (e) 32x32 merged blocks (f) 64x64 merged blocks

Figure 3-6: Quadtree block generation using the example shown in Figure 6 in [12] with $L = 4$, $K = 2$, $CF = 0.2$ and $T = 3$.

image increases, the computational speed for its bottom-up scheme does not increase significantly and still is faster compared with **CBM**.

3.4.2 Dominant Cloud Motion Patterns

To suppress image noise and neglect falsely estimated motion vectors, we propose to identify the dominant patterns of cloud motions and use them to refine the entire motion field. First, appropriate image preprocessing is necessary to undistort the TSI images ensure that the majority of cloud motions obtained from undistorted image are simple and translative. Subsequently, we can apply a straightforward clustering (e.g. k-means) to effectively group them, to find dominant patterns, and to remove abnormal ones [132] [12].

However, the clustering approach does not take into consideration the weight of each motion vector, i.e., the size of actual cloud block, and thereby, might

miss some dominant vectors. In this paper, we utilize the histogram statistics presented in [133] to extract the N most frequent cloud motions in a sky image, and use them to correct and refine the results of estimated motions. To accurately determine the significant motion modes for cloud pixels and exclude the small-scale motions caused by image noise or sky pixels, we only consider those *obvious* motion vectors that have adequate velocity between two consecutive frames ($\|u\|^2 + \|v\|^2 \geqslant 1$). Thereby, stationary sky pixels or slow-moving clouds are ignored in calculating dominant motions.

In our implementation, we first produce a dense motion field based on the motion vectors detected in previous block matching. Each cloud pixel acquires the associated motion vector from the cloud block on which it resides. Thereafter, we create a 10×10 2-D histogram for the motion vectors of all cloud pixels. This design considers the size of a cloud block, and a large block has many pixels, each of which contributes to the count, viz., the weight of the shared block-wise motion vectors to the histogram. Then we picked the N most-frequent motion modes, and grouped the pixel-wise motion vectors into N groups. For each motion group, we selected the median motion vector as the dominant motion pattern, and thereby, generated a collection of dominant motion vectors ($\Omega = \{(u_i, v_i) | i = 1, 2, ...N\}$). Finally, we refine all non-zero motion vectors by resetting their values to the closest dominant motion vectors: $(\tilde{u}, \tilde{v}) = \underset{(u_i, v_i)}{argmin} \left(\|u - u_i\|^2 + \|v - v_i\|^2\right)$ where $(u_i, v_i) \in \Omega$. Here, we use $(\tilde{u}, \tilde{v})$ as the reference vector to guide the refinement of optical flow in the next section.

3.4.3 Context-Aware Variational Model and Its Refinement

The original OF models lack of the contextual information and are agnostic to the existing domain knowledge, and consequently, they often assign erroneous flow vectors to cloud pixels and are sensitive to noises in images. To cope with those issues and utilize the information of motion layers and cloud distribu-

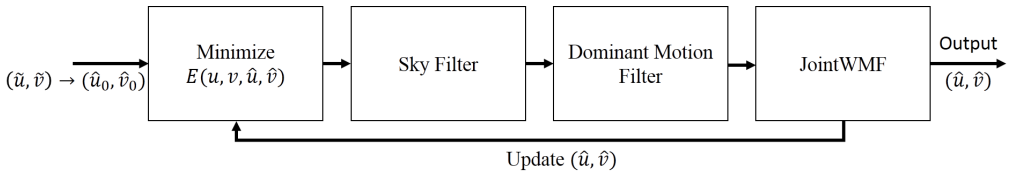


Figure 3-7: Minimization of new variational model.

tion, we proposed a new OF model. First, we revise the energy-like objective to assimilate the aforementioned dominant motion patterns, and to use them to create a context for calculating the optical flow at each cloud pixel. However, a motion field obtained only from the straightforward process of energy minimization still suffers those issues. The general practice in many prior works is to apply post-processing techniques, such as median filtering [134] or signal-noise-ratio threshold [80] to mitigate the impact of image noise and remove the outliers. Therefore, we also follow the practice and apply three filters thereafter to further assimilate cloud information, utilize dominant motion patterns, and remove noise in the motion field. Given the facts that one iteration is not sufficient to correct the motion-field and the reference vector still needs to be refined to absorb new information and thereby to closely represent the current motion field, we design an iterative algorithm to generate and optimize the dense flow field and to update the subsequent reference motion vectors. This design recognizes that the vectors detected by the BM and the OF models are actually inter-dependent. For the optimal results, they should be integrated into the same framework to ensure them to mutually enhance each other.

Inspired by the two-stage (assimilation and updating) alternative minimization framework in [127], we propose a new OF model based on a four-step optimization pipeline that iteratively generates a dense flow field, as shown in Figure 3-7. In this four-step framework, we iteratively solved and refined the OF result via introducing the auxiliary motion field $(\hat{u}, \hat{v})$ to: (1) estimate the flow field (u, v) so that it is nearly identical to $(\hat{u}, \hat{v})$, and (2) calculate the best $(\hat{u}, \hat{v})$ using three motion filters on the current tracking result (u, v).

The first step of the pipeline in Figure 3-7 is to formulate the OF objective $E(u,v)$ to be a summation of four energy terms as follows:

$$\begin{aligned} E(u,v) = E_{color}(u,v) + \lambda E_{smooth}(u,v) + \\ \gamma E_{grad}(u,v) + \beta E_{motion}(u,v,\hat{u},\hat{v}) \end{aligned} \tag{3.8}$$

where E_{color} and E_{grad} are defined in Eqs. 3.6 and 3.7. E_{smooth} is identical to the second term in **BA** (Eq. 3.4). Here, we add a new energy term E_{motion} so to encourage the optical flow to be identical to the reference/auxiliary motion field $(\hat{u},\hat{v})$, whilst allowing a small perturbation from it, and meanwhile penalizing any large deviation. E_{motion} is defined as follows:

$$E_{motion}(u,v,\hat{u},\hat{v}) = \sum_{x,y} \rho(\|u-\hat{u}\|^2 + \|v-\hat{v}\|^2) \tag{3.9}$$

The motion perturbation presented in [132] can track non-rigid transformation and deformation on the local scale. By setting the initial reference motions to be the dominant motion patterns, i.e., $(\hat{u}_0,\hat{v}_0) = (\tilde{u},\tilde{v})$, for each pixel, we constrain clouds to move along the major trend (wind direction) while permitting a small deviation to capture the local deformation and the changing shape of the clouds.

Optimizing Eq. 3.8, however, is a non-trivial task since the original problem is not linear. Its minimization requires a two-step linearization based upon the corresponding Euler-Lagrange equations, and successive over-relaxation (SOR) [135]. The details of discretization and implementation are given in [73] and [136].

However, the motion field obtained by minimizing the energy-like objective still contains anomalies that do not match the real cloud movements. The general practice in many previous efforts was to apply post-processing techniques, e.g. median filters. In the pipeline in Figure 3-7, we use three different filters (sky filter, dominant motion filter, and weighted median filter) thereafter

for assimilating cloud information, utilizing dominant motion patterns, and de-noising the motion field.

Sky Filter processes the clear sky pixels that are identified by velocity threshold. We assume that the clear sky regions in images do not move between consecutive frames, and set the motions of all sky pixels that acquire a small-scale motion (i.e. $u^2+v^2 \leq T_{sky}^2$) in the previous step to be *zero* in both the x and y directions. It eliminates the small motion vectors caused by image noise and avoids accumulating errors in the iterative minimization step. It is noteworthy that instead of considering all clear sky pixels, we filter out only the ones with a small movement based on the threshold criterion T_{sky} because cloud pixels in thin clouds or the images containing a high level of image noise have a high probability of being falsely categorized into the class of clear sky. Consequently, relying on cloud mask only potentially neglects these special cases, and accidentally removes the prominent motions of these mis-classified pixels.

A Dominant Motion Filter is designed to identify the outliers with reference to the dominant motion patterns. If a motion vector at position (x,y) has significant deviations from all dominant motion patterns (Ω), then we identify it as an outlier, i.e., if a motion vector meets the following condition, we will remove it.

$$\left\{(u_{x,y},v_{x,y})|(u_{x,y}-u_i)^2+(v_{x,y}-v_i)^2>\varepsilon^2,\forall(u_i,v_i)\in\Omega\right\} \tag{3.10}$$

This filter significantly helps to refine the motion field for the next round of minimization and to update the reference motion field.

A Weighted Median Filter is widely adopted to smooth and de-noise the motion field in the post-processing step [127]. Our model employs this filter to update reference (auxiliary) motion field $(\hat{u},\hat{v})$. The weighted median filter can be calculated as follows:

$$\min_{\hat{u}_{x,y},\hat{v}_{x,y}} \int_{(x',y')\in\Gamma(x,y)} w_{x,y}^{x',y'} \left(\left| \hat{u}_{x,y} - u_{x',y'} \right| + \left| \hat{v}_{x,y} - v_{x',y'} \right| \right) \tag{3.11}$$

where $\Gamma(x,y)$ is the set of (x,y)'s neighborhood pixels (x',y') within a predefined window (e.g. 5 x 5) centered at (x,y). $w_{x,y}^{x',y'}$ stands for the weight of the affinity (similarity) between two pixels. In our implementation, we approximate its calculation with the color difference in an image:

$$w_{x,y}^{x',y'} = e^{-\frac{\|F(x,y)-F(x',y')\|^2}{2\sigma^2}} \tag{3.12}$$

where F represents the color image with the R, G, B channels. To compute w efficiently between each pair of pixels, we adopt the joint-histogram weighted median filter (JointWMF [137]) to reduce the execution time of updating $(\hat{u},\hat{v})$. Compared with the weighted median calculation in [127], the JointWMF significantly speeds up the running time by more than 10 times and at the same time preserves the quality of estimation [137]. Afterward, we pass the updated $(\hat{u},\hat{v})$ for the next round of the iterative minimization of $E(u,v)$. We iterate the process three times to produce the final output of the dense optical flow field.

3.5 Experiments

To validate the effectiveness of our proposed model, we propose establishing a comprehensive simulation framework to incorporate translative cloud motions, cloud deformation, and various levels of noises into the synthetic sky images. Then, we can evaluate our model under these circumstances with the known ground truth. Furthermore, we will apply our model to real images collected from various imaging systems and validate its performance in real-world applications.

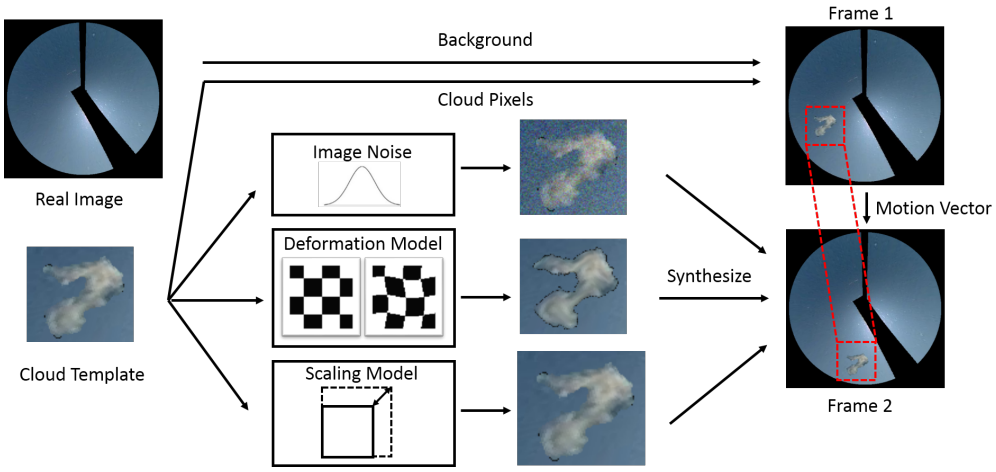


Figure 3-8: Framework of synthesizing two-frame sequence of sky images.

3.5.1 Simulated Dataset of Cloud Images

To evaluate the robustness of tracking models, we generated three different two-frame sequences of synthetic images from a cloud template (foreground) and a real sky image (background), each of which represents image noises or a different type of cloud motion, i.e., deformation or scaling, as shown in Figure 3-8. We first overlaid the template on the background image to create the first synthesized frame. Then, we employed different transformation models or injected image noise to generate a new template, and subsequently applied a translative motion vector to synthesize the second frame. Consequently, the ground-truth motion field is a combination of the template translation and the motion perturbations originating from the deformation or scaling models. We detail the three types of simulations as follows:

1) Simulation with a deformation model. Inspired by prior work [77] that simulated the changes in the Magnetic Resonance Image of brain (MRI) by a 2-D synthetic deformation model applied on a template, we utilized the same model of image distortion to resemble the changes in shape and deformation of clouds during their movements between two consecutive frames; we also adopted their spatial sinusoidal functions with the same settings to the deforma-

Table 3.2: Overview of seven sky image datasets.

Dataset	Instrument	Range	$image_{\#}$	Rate	Res_{raw}	Res	FOV
BNL1	TSI 880	2013-05-07 $\sim$ 2013-06-25	562	10s	480x640	500x500	120°
BNL2	TSI 880	2013-05-07 $\sim$ 2013-06-25	562	10s	480x640	500x500	120°
BNL3	TSI 880	2013-05-07 $\sim$ 2013-06-25	562	10s	480x640	500x500	120°
SGPC1	TSI 440A	2012-04-01 $\sim$ 2012-06-15	1680	30s	480x640	500x500	120°
TWPC1	TSI 440A	2010-08-25 $\sim$ 2010-11-01	1654	30s	480x640	500x500	120°
HD-SI	IP-cam	2014-10-01 $\sim$ 2014-11-01	434	10s	1920x1920	1000x1000	140°
HD-TSI	IP-cam	2014-09-01 $\sim$ 2014-10-01	434	10s	2592x1920	1000x1000	140°

tion parameters of amplitude (A) and spatial period (S).

2) Simulation with scaling. We utilized an image re-sizing tool to simulate the cloud's shrinking and expansion. The size of the new template is the scale ratio (SR) multiplied by its original size. In our experiment, we set SR in the range between $(0.9 \sim 1.2)$ with a step size equal to 0.1.

3) Simulation with different levels of noises. Following the idea proposed in [51] that added the Gaussian noise to the images of carotid arteries, we also injected the same type of random noise into the R,G,B channels of our synthetic image sequences. In this simulated dataset, the signal-to-noise ratio (SNR) ranged between 20 to 50.

3.5.2 Real Sky Imagery Datasets

In this paper, we use seven types of sky images that were collected from different locations, and/or different types of sky cameras (See Table 3.2 for details). The first three datasets, denoted as *BNL1*, *BNL2*, and *BNL3*, were obtained from a small network of three TSIs deployed in the Long Island Solar Farm (LISF) at Brookhaven National Laboratory. These TSIs have an identical configuration, and are placed close to each other so to acquire the overlapping views for the purpose of tracking the micro-scale cloud motions. To cover the representative conditions of various types of weather and clouds, we chose 562 timestamps from May/07/2013 to June/25/2013 that are synchronized among all three TSI datasets. We converted the raw TSI images (480 x 640) into their planar views

(a) TSI Original (b) HD-TSI Original (c) HD-SI Original

(d) preprocessed (e) preprocessed (f) preprocessed

Figure 3-9: Raw image and preprocessed view of TSI, HD-TSI, and HD-SI.

to eliminate distortion in the images and cropped the resulting planar images with a resolution of 500 x 500 pixels, and a pre-defined Field of View (FOV) of about 120° to remove low-quality regions. Lastly, we masked out the irrelevant pixels in the areas with artifacts, such as the camera supporting arm, the shadow band, the out-of-FOV areas, and ground obstacles, as illustrated in Figures 3-9a and 3-9d.

The fourth and fifth dataset are from the TSIs in different locations: *SGPC1* represents the dataset that consists of a pair of two image frames collected daily every 30 minutes between 10:00AM to 16:00PM (local time) from April/15 to June/01/2012 from the central facility of the Southern Great Plains; *TWPC1* is collected from the Tropical Western Pacific site in Manus Island. Similar to *SGPC1*, *TWPC1* comprises the pairs of consecutive frames every 30 minutes daily from August/25 to November/01/2010. We also preprocessed the raw TSI images of these two datasets with the same method to correct distortions, and the same mask with a FOV of 120°. These two datasets ensured that our proposed

model of estimating cloud motions has sufficient coverage for various cloud- and weather-conditions in different geographical locations.

The remaining two datasets were obtained from the high-definition cameras deployed at Brookhaven National Laboratory. The first dataset, *HD-TSI*, is from a high definition TSI in which we replaced the built-in camera by a security camera with a high-resolution of 2592 x 1920 pixels, (Figure 3-9b). After preprocessing, we generated a planar view of 1000×1000, with a FOV of about 140° (Figure 3-9e). The other HD-image dataset, *HD-SI*, was obtained from a security camera that faced upward to take pictures of local sky regions directly via its fish eye lens (Figure 3-9c). After preprocessing, we also retained the images with a resolution of 1000 x 1000 and the same FOV of about 140° (Figure 3-9f). These two datasets helped us to validate whether our model is scalable to different types of cameras and large images.

3.5.3 Evaluation Metrics

We adopted five metrics to quantitatively evaluate the performance of our proposed model of estimating cloud motions and to compare it with several representative models. Here, we denote the estimated motion as $\mathscr{W}_{(x,y)} = (u, v)$ and the ground-truth motion as $\mathscr{W}^{\mathscr{G}}_{x,y} = (u_g, v_g)$. Because a dense motion field is our focus, we will evaluate the average performance of motion tracking on all P pixels where P is the total number of pixels in images.

1) Optical-Flow Color Map. We propose using a color map to visualize a dense motion field, a common approach adopted by computer vision. With it, pixel-level motion vectors are normalized and represented by different colors based on their magnitude and orientation. In this paper we normalize all motion vectors to be within a square ranging between (-15,-15) to (15,15), and thereafter assign each pixel a color code from the motion color map in Figure 3-10.

2) Average angular error (AAE). The AAE measures the errors that arise

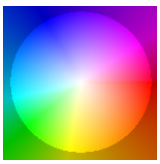


Figure 3-10: Color coding scheme for the motion vectors in the range between (-15,-15) to (15,15).

from the angular deviation of an estimated motion vector from the corresponding ground-truth orientation, and is calculated via the following equation:

$$AAE = \frac{1}{P}\sum_{(x,y)} arccos\left(\frac{\mathscr{W}_{(x,y)} \circ \mathscr{W}^{\mathscr{G}}_{(x,y)}}{\|\mathscr{W}_{(x,y)}\|\|\mathscr{W}^{\mathscr{G}}_{(x,y)}\|}\right) \tag{3.13}$$

3) Standard deviation of angular error (STDANG). STDANG denotes the standard deviation of the angular errors and is defined as follows:

$$STDANG = \sqrt{\frac{1}{P}\sum_{(x,y)} \left(arccos\left(\frac{W_{(x,y)} \circ \mathscr{W}^{\mathscr{G}}_{(x,y)}}{\|\mathscr{W}_{(x,y)}\|\|\mathscr{W}^{\mathscr{G}}_{(x,y)}\|}\right) - AAE\right)^2} \tag{3.14}$$

4) Average end-point error (AEPE). This evaluates the average deviation of motion vectors from the ground-truth vector.

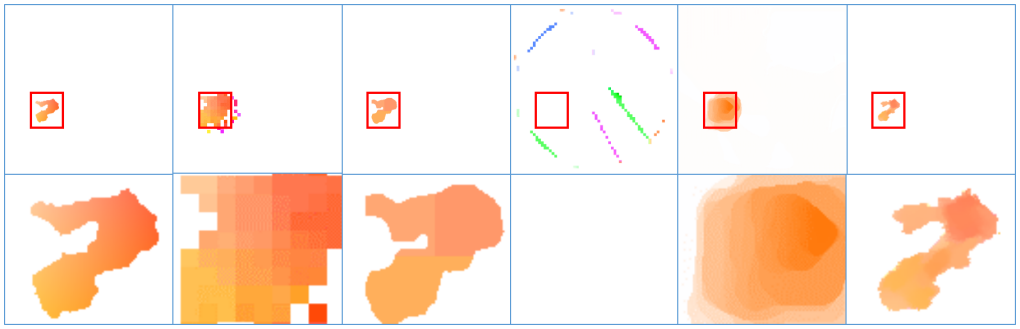
$$AEPE = \frac{1}{P}\sum_{(x,y)} \|(\mathscr{W}_{(x,y)} - \mathscr{W}^{\mathscr{G}}_{(x,y)})\| \tag{3.15}$$

5) Mean Absolute Error (MAE). The MAE is used to calculate the mean absolution error between the ground-truth image (I_g), and the predicted one. Here, the predicted image ($I_{Predict}$) is subsequent to the current image frame in the same sequence and is generated by applying the estimated motion vectors to all pixels thereby mapping those from the current image frame to the next frame. We generally use the MAE score to evaluate the performance of motion estimation [47] when the ground-truth for motions does not exist.

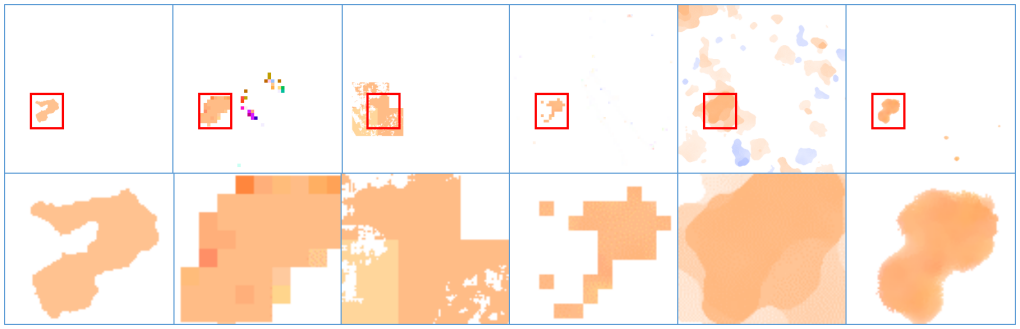
$$MAE = \frac{1}{P} \sum_{(x,y)} \left| I_{Predict}(x,y) - I_g(x,y) \right| \tag{3.16}$$

3.5.4 Experiments on the Sequence of Simulated Image Frames

(a) Deformation ($A = 2$, $S = 64$)



(b) Scaling ($SR = 1.2$)



(c) Image Noise ($SNR = 20$)

Figure 3-11: The color map of the results of all tracking models on three types of synthetic images. *Left to right*: Ground-truth motion, **BM**, **CBM**, **MPIV**, **LDOF**, and **Hybrid**.

We applied seven classic models and our proposed hybrid model to estimate cloud motions in the three types of simulated image-sequences. Figure 3-11

Table 3.3: The AAE, STDANG, and MEPE scores of the motion-tracking models on simulated images with various deformation settings.

	A=1,S=16			A=3,S=32			A=5,S=32		
	AAE	STDANG	MEPE	AAE	STDANG	MEPE	AAE	STDANG	MEPE
BM	1.771	11.61	0.162	1.692	11.16	0.175	1.705	10.99	0.186
MPIV	0.956	7.710	0.057	0.985	7.829	0.071	1.262	8.934	0.101
CBM	0.368	4.967	0.038	0.501	5.499	0.063	0.660	6.583	0.088
HS	1.573	8.048	0.075	1.551	7.947	0.076	1.572	8.380	0.089
LK	5.277	16.91	0.267	5.456	17.28	0.299	5.761	18.08	0.360
LDOF	0.628	6.671	0.043	0.516	5.393	0.041	0.547	**5.401**	0.061
BA	0.488	5.587	0.038	0.584	5.500	0.054	0.802	7.061	0.074
Hybrid	**0.218**	**3.852**	**0.021**	**0.324**	**4.435**	**0.035**	**0.494**	5.655	**0.058**

Table 3.4: AAE, STDANG, and MEPE scores of motion tracking models on simulated images with three levels of Gaussian noise.

	SNR=20			SNR=30			SNR=40		
	AAE	STDANG	MEPE	AAE	STDANG	MEPE	AAE	STDANG	MEPE
BM	1.813	12.03	0.216	2.355	13.77	0.278	3.271	15.05	0.279
MPIV	0.707	6.582	0.044	0.772	6.427	0.043	1.771	8.710	0.073
CBM	4.097	17.75	0.339	5.003	20.22	1.891	1.034	8.968	0.185
HS	0.637	6.104	0.055	0.715	6.617	0.056	0.799	6.977	0.058
LK	31.17	34.14	1.303	23.95	32.74	1.048	15.51	28.56	0.662
LDOF	33.14	38.04	2.374	2.032	11.95	0.120	1.052	9.026	0.080
BA	16.67	29.28	0.707	6.050	19.33	0.254	1.331	10.09	0.092
Hybrid	**0.233**	**4.225**	**0.027**	**0.231**	**4.224**	**0.024**	**0.231**	**4.241**	**0.023**

shows the output motion fields in the format of color map that correspond to the ground-truth, **BA**, **CBM**, **MPIV**, **LDOF**, and **Hybrid** respectively. The first row of images in Figure 3-11 indicates the detected motions of all pixels in the entire simulated image whilst the second row is the zoom-in view of the motions of the cloud region (bordered in red). From Figures 3-11a, 3-11b, we observed that our hybrid model can accurately detect the deformations of clouds and generate the best motion color map, in particular, in the areas around the edges, to closely resemble the ground-truth, when compared with other models. As shown in these two figures, even though **CBM** can estimate the major trend of motions based on its intelligent detection of big blocks of clouds, nevertheless it neglects the underlying deformations in shape within each block, and fails to capture the local variations in the area of clouds. Consequently, its color map appears to be in a solid color with the obvious artifacts arising from the decomposition of a

cloud block.

Figure 3-11c shows the third case of noises in images. Here, each image is corrupted with the Gaussian noise at a level of $SNR = 20$. We observed that because of the existence of these noises, most models fail to recover shape of the cloud and produce falsely estimated motions for those stationary pixels. With the help of the filters and the dominant patterns of motion, our proposed hybrid model preserves a reasonable segmentation between the motion fields of a cloud region and the area of clear sky, whilst incurring only a much smaller number of outliers in the area of clear sky than do the remaining models.

Tables 3.3 and 3.4 show the performance number of all the models with regard to two different deformation settings and multiple levels of noise. We conclude that the *Hybrid* model consistently outperforms other models in most cases in terms of three different metrics, i.e., AAE, STDANG, and MEPE. For the most difficult case of simulating deformation (i.e. $A = 5$, $S = 32$), **Hybrid** retains the lowest level of errors in terms of AAE and MEPE and the second to the lowest in terms of STDANG, which is only 5% higher than the best score obtained by the *LDOF* model.

3.5.5 Experiments on TSI and High Definition (HD) Images

To validate the accuracy of detected motion vectors, we applied the resulting motion field between two consecutive frames (t-1 and t) to predict the next frame (t+1) and to compare it with the true image. In Figure 3-12, the first row lists the raw data of three frames at time $t-1, t, t+1$ (ground-truth) and a motion mask to remove outliers and calculate the MAE score. The second row presents the motion fields of **BM**, **MPIV**, **BA**, and **Hybrid** respectively. The third row shows the predicted images using detected motions and image interpolation. The fourth row displays the color maps of the difference between the predicted images by the four models and the ground-truth frame at time $t+1$. We observed

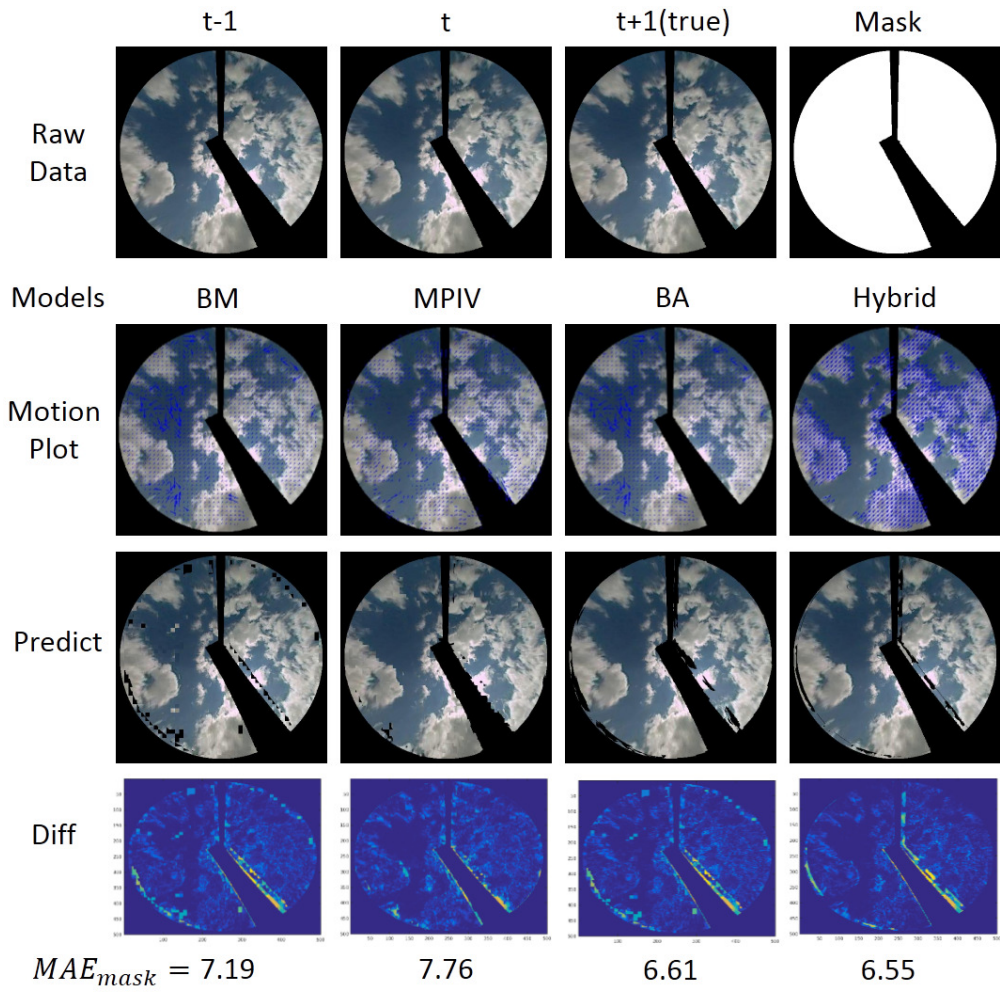
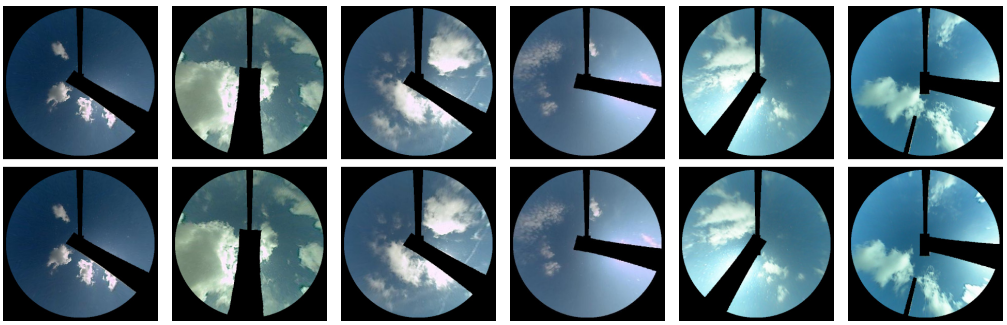


Figure 3-12: Using the estimated motion fields to predict the next image frame at time $t+1$.



(a) Scatter (b) Green (c) Fast (d) Multi (e) Deform1 (f) Deform2

Figure 3-13: Six cases selected from the TSI datasets. (a) Scattered clouds; (b) Images from *BNL*1 dominated by a green color; (c) Fast-moving clouds in dataset *BNL*3; (d) Multi-layer clouds; (e) and (f) Images from two different TSI sites (*SGPC*1 and *TWPC*1), both showing fast motion of clouds and deformation in their shape.

that our **Hybrid** model can generate stable motion vectors near the boundary of each piece of cloud. In contrast to other models, **Hybrid** is able to capture the small changes of clouds and produce less MAE errors in texture-rich regions.

In our experiment, we selected eight representative cases to encompass various cloud conditions that appeared in different sites and sky imagers. The first six of them shown in Figure 3-13 are in TSI datasets: Figure 3-13a shows the condition of scattered clouds whose boundaries are clearly segmented. All pieces of cloud have negligible deformations and identical motion vectors. Figure 3-13b shows an abnormal image with a different color scale by an artifact of the camera's setting and dominated by green. Figure 3-13c contains fast-moving clouds with a velocity of approximate $(u,v) = (-17,2)$. Figure 3-13d comprises two layers of clouds, each with different motion vectors: the first one has $(u,v) = (6,1)$ whereas the other has $(u,v) = (11,-6)$. This example targets at evaluating each model's capability to determine the multiple layers of cloud motion. Figure 3-13e and Figure 3-13f, respectively, come from the dataset of *SGPC*1 and *TWPC*1. The clouds in these two examples move quickly between the frames, and more importantly, involve local deformations, such as cloud dissipation and changing shape. The average translative movement in the first

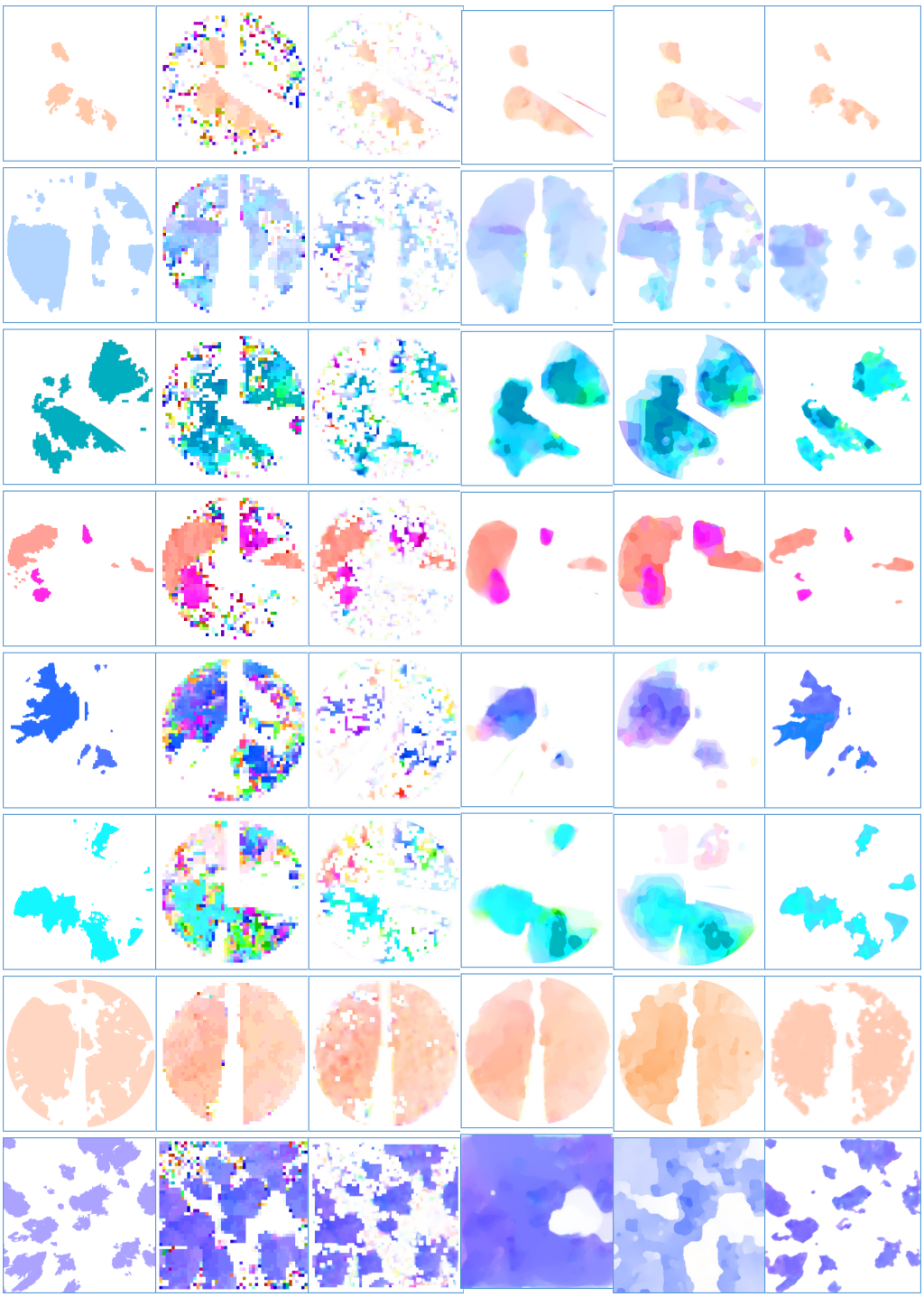
Table 3.5: MAE score of models of eight selected cases

	Scatter	Green	Fast	Multi	Deform1	Deform2	Fig. 3-9e	Fig. 3-9f
BM	19.6	23.5	39.4	33.9	54.9	54.5	13.1	19.6
MPIV	7.8	19.1	29.0	15.6	34.8	44.4	16.9	18.6
BA	3.7	11.4	18.2	6.4	20.3	25.1	12.4	10.7
Manual	3.0	8.0	15.0	4.0	18.0	22.0	8.8	7.3
Hybrid	3.3	8.2	15.4	4.6	18.5	22.5	9.3	8.8

example is close to $(-25, -19)$, and the average motion during half a minute in the second one is about $(-39, 10)$. The remaining two cases are from the HD datasets presented in Figure 3-9e and 3-9f.

Because there is no existing ground truth for cloud motions in a real-world application of sky image, we manually generate one from images, and use it as the ground-truth for evaluating models. First, we manually annotate clouds in these selected images and estimate their average translative movement across two frames (the first and second rows of images in Figure 3-13) for each piece of cloud. Then, we generate the color maps from the manually annotated ground truth (termed **Manual**) and the results of the motion estimation models, including **BM**, **MPIV**, **BA**, **LODF**, and **Hybrid** (Figure 3-14). Lastly, we apply the resulting motion field to the first image frame so to predict the subsequent frame and calculate the MAE score between the predicted frame and the second frame (Table 3.5). We observed that **Hybrid** attains the best approximation to the manually annotated color map. More importantly, **Hybrid** is robust in all of the eight representative cases and has the fewest MAE errors among the five models.

Table 3.6 summarizes the MAE score of all the motion estimation models over different real image datasets. It shows that the **Hybrid** model outperforms the remaining classic models for the TSI and HD imager datasets. Compared with the best-performing block matching approach, **CBM**, and the most accurate OF method, **LDOF**, our proposed hybrid approach further cuts down the overall error rate by more than 30% and 15%, respectively.



(a) Manual (b) BM (c) mpiv (d) LDOF (e) BA (f) new

Figure 3-14: Color map on eight selected cases.

Table 3.6: MAE score of motion tracking models on seven real images

	BNL1	BNL2	BNL3	TWP	SGP	HD-SI	HD-TSI	Overall
BM	20.4	18.0	24.1	34.6	39.0	18.8	15.7	24.4
MPIV	20.3	15.3	23.6	26.1	27.5	19.6	15.0	21.1
CBM	7.9	5.0	8.7	11.9	15.6	13.4	5.8	9.8
HS	13.5	7.8	11.1	17.8	16.3	18.8	10.9	13.7
LK	42.9	38.4	48.4	51.9	56.1	40.2	39.9	45.4
BA	9.8	6.6	10.9	11.6	11.5	12.5	7.6	10.1
LDOF	7.9	5.0	7.0	11.1	9.8	9.3	5.7	8.0
Hybrid	6.2	4.2	4.7	9.8	8.8	8.7	5.3	6.8

3.5.6 Experiment Findings and Discussion

The experiment analysis on cloud pixel identification shows that the proposed cloud detection pipeline was robust under various cloud/lighting conditions and obtained an overall accuracy of 96.6% for detecting clouds. It verifies that the image features extracted at local region, i.e., LOF and STD, are effective for gathering neighbor information and defining the edges of cloud area. This result further validated that our multi-source pipeline design, in particular, the incorporation of images streams from multiple cameras synchronized with each other and the following histogram equalization to eliminate artifacts of individual cameras, not only improves the accuracy of the cloud classifier, but also increases the stability of cloud detection over various complex cases involving anomaly. When we tested our pipeline on a different type of imager, a Webcam in Figure 3-3, the cloud detection pipeline still identified the majority of cloud pixels where $AC_{cld} = 96.2\%$.

Furthermore, we experimented on cloud motion tracking with various image datasets, and confirmed that our new model is computationally efficient in cloud block matching and in general, more than five times faster than state-of-the-art BM methods. It coincides with the fact that our QCBM module effectively segments cloud regions into a hierarchy and optimizes the number of generated cloud blocks starting from a low level and via a bottom-up merging scheme. By comparing our hybrid model with state-of-the-art models over synthetic image

datasets, we confirmed that the auxiliary motion energy term enhanced the hybrid model with resilience to noise and capability of tracking complex shape deformations that are absent from both the OF models and block-matching methods. Consequently, the hybrid model generated motion color maps with a much better accuracy than all other tracking algorithms selected for comparison. As we expected, the OF methods experienced significant performance degradation for fast-changing clouds in certain selected cases. It agreed with the observation that traditional OF approaches are ineffective with regard to detecting large movements and sensitive to the initialization settings [73]. To solve this problem, our hybrid model utilizes dominant motion layers and a four-step minimization framework. In our experiment, we generated color maps to represent and visualize the motion vector field, and found that the motion layers and customized filters in our hybrid model help significantly reduce abnormal motion vectors, especially around the edges of cloud regions.

Our experiment also demonstrated that motion vector clusters and succeeding cloud classification based on the clustering results further improved cloud movement detection. It also confirmed the robustness of a well-designed weight median filter in the alternating minimization algorithm [127]. In summary, the experiment results vetted each element of our minimization pipeline, and collectively, the four-step alternating minimization effectively ensures the performance of de-noising and minimizes the outliers in the resultant motion field.

3.6 Summary

In this chapter, we focus on the problem of cloud detection and tracking in different types of sky images. We discuss our investigation of state-of-the-art motion estimation methods for tracking and predicting motion of clouds, and quantitatively analyze classic models of both block matching and optical flow using various sky imagers and synthetic images. By taking advantage of all

prior models, we propose a new hybrid model to employ cloud mask and Quadtree in block-wise motion-tracking, to extract the dominant motion patterns via histogram statistics, and to estimate a dense motion field at pixel level via customized motion filters and objective function.

To validate the effectiveness of our model for cloud motion estimation, we not only collected various types of real images from different sites, but also synthesized three image-sequences to simulate the shape-changing clouds and image corruption with random noise. The results show that compared with classic models, our new model can accurately capture cloud deformation and is resilient to image noise. It consistently produces the best color map and greatly reduces the angular error (AAE) by at least 30% in all simulated image sequences. In terms of real images, the new model consistently extracted clear motions of clouds under various conditions (e.g., multiple layers and fast-moving clouds) and lessened MAE by at least 10% for both TSI and HD image datasets.

Chapter 4

Multi-TSI Integration

In this chapter, we present our research on multi-camera integration for 3D cloud tracking and solar forecasting. A single TSI does not supply three-dimensional spatial information of cloud that is necessary for regional forecast. To resolve this problem and extract the necessary spatial information of clouds, we propose to use a three-TSI network in which each camera has overlapping FOV with the other two and observes cloud simultaneously. Based on the real-time streams of multiple synchronized sky views from three TSIs, we investigate the spatio-temporal correlations among nine frames (3×3 along the temporal axis $t-2, t-1, t$, and the spatial axis: Cam_1, Cam_2, Cam_3) and thereby determine cloud base height and motion vectors at each time stamp. Afterward, we apply clustering algorithm to group cloud motions into cloud layers and anomaly detection to denoise motion field. Once the estimated cloud movements are obtained, we can predict cloud presence in the future. Subsequently, we design multiple machine learning methods and choose effective input image features for short-term solar forecast.

4.1 Introduction

As single imager has limited FOV and is unable to detect clouds' vertical distribution, prior studies cannot accurately predict sun-occlusion effects and are often relying on other instruments (e.g. LIDAR) to provide additional information such as cloud base height. To solve this issue, we propose to use synchronous multi-camera network to observe clouds from different view angles, and consequently track clouds in 3D space. In detail, we propose employing a novel system to identify the primary cloud layers and extract the effective image features that are relevant to forecasting the surface radiation. Our contributions are summarized as follow:

1. **New Cloud Detection Methodology:** We propose an effective cloud detection pipeline for classifying cloud pixels and utilizing multiple TSIs to correct erroneous TSI images or compensate for abnormal exposure.

2. **Multiple-Source Cloud Tracking:** In contrast to the related work described in this section, we design a novel tracking algorithm to incorporate the spatial and temporal correlations between clouds among multiple TSIs and time frames.

3. **Multi-layer Recognition:** We develop a clustering-based technique to integrate the results of individual block-wise tracking into multiple cloud layers, each of which forms a major wind field.

4. **Feature Selection:** Instead of tracking a single cloud pixel, we use sky images to generate both global features at the image level as well as local variations within a small pixel 7 x 7 window.

5. **New Regression-based Irradiance Forecast Models:** We propose four regression-based solar irradiance models that utilize the predicted features of cloud pixels, and compare these models with the persistent model and its variant based on Support Vector Regression.

6. **High Accuracy in Field Studies and Validation:** We confirm, in our

preliminary analysis, that the proposed system for detecting and tracking clouds and predicting their relevant features can accurately forecast solar radiation for up to 15 minutes.

4.2 Multi-TSI Network Setup

4.2.1 Instruments

(a) TSI_1 (b) TSI_2 (c) TSI_3

(d) Google Maps View

Figure 4-1: (a), (b), and (c) are the preprocessed views of the three TSIs. (d) is the Google Maps view of the three TSIs from left to right (camera icons) and the 25 solar radiation sensors (red). The distance between TSI_1 and TSI_2 is 2477 meters, and that between TSI_2 and TSI_3 is 956 meters.

The data used in this paper were obtained from the Long Island Solar Farm

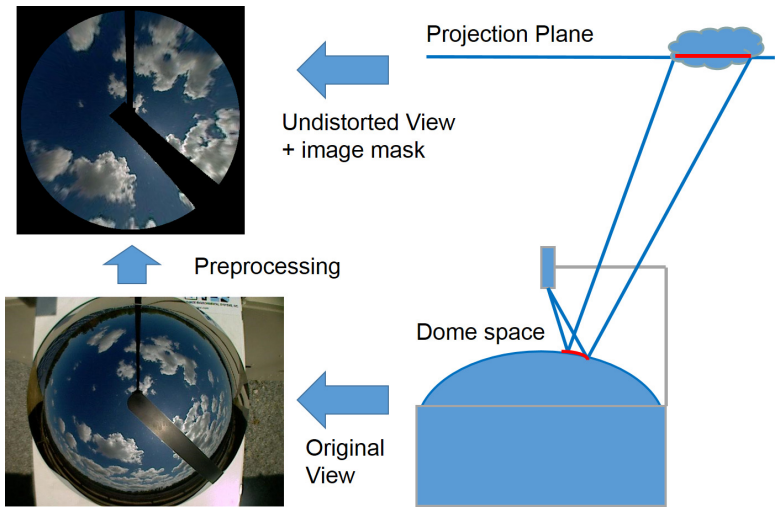


Figure 4-2: Procedure for preprocessing the TSI images. The original image is undistorted from the original dome space to the planar space via coordinate transformation. The output image is cropped based on a pre-defined FOV range and masked to remove irrelevant areas, such as the supporting arm and the shadow band.

(LISF), a 32-megawatt solar photovoltaic power plant built by a collaboration between BP Solar, the Long Island Power Authority (LIPA), and the Department of Energy. The LISF, located at the Brookhaven National Laboratory, is currently the largest solar photovoltaic power plant in the Eastern United States. The cloud tracking system (Figure 4-1) consists of a network of three total sky imagers (TSI_1, TSI_2, and TSI_3). Their positions are triangulated to ensure good coverage of the sky above the solar farm. TSI_1 collects sky images of a region near the solar farm. TSI_2 is deployed in the middle of the farm to ensure sufficient overlap of its views with those of the other two TSIs. TSI_3 is located in the northern area of the solar farm. We installed all three TSIs at the same altitude and tuned their orientations to ensure that their camera-supporting arms point north. Consequently, the supporting arm in each TSI images overlaps with the vertical(y) axis on the image which is aligned toward the north.

All TSIs uploaded real-time video streams with a raw resolution of 640x480, and at a streaming rate of 10 seconds per frame to a centralized database server.

We then applied preprocessing techniques, as illustrated in Figure 4-2, to undistort the raw images and remove unrelated artifacts, i.e., the shadow band and supporting arm. The primary advantage of preprocessing the images, as described in our previous work [21], is that the planar view obtained from the coordinate transformation of the raw images is more effective for estimating cloud motions. In this study, we projected the original coordinates to a flat plane with a given resolution (500x500 pixels) and cropped the planar image using a pre-defined view angle range (zenith range $0^\circ - 60^\circ$) to ensure an FOV of approximately 120°. In addition, because the camera-supporting arm may not be exactly aligned with the vertical direction in a TSI image, we adjusted the orientation in the preprocessing step by rotating the images by several degrees. To simplify the coordinates transformations between different TSIs, we unified the conditions and specifications of all TSI devices. Because each TSI has an identical FOV and resolution, all the preprocessed images reside on the same projection plane, wherein pixels are distributed evenly. Finally, we applied a pre-calculated mask to remove irrelevant areas, such as the supporting arm and shadow band. The preprocessed image was then generated, with all irrelevant pixels being set to black (Figures 4-1a, 4-1b, and 4-1c).

We deployed 25 pyranometers in the LISF, as shown in Figure 4-1d, to measure the surface solar irradiance. These sensors measure the global horizontal irradiance (GHI) in real time. The measurements which are recorded every 10 seconds, are synchronized with the TSI observation. The variations in zenith and the diurnal and seasonal patterns are also recorded in the raw GHI measurements, and therefore bias our subsequent irradiance forecasting models. To mitigate this potential bias, we normalized each radiation value to a clear-sky index k^t during model training and testing. Letting GHI^t be the raw GHI measured at time t and GHI^t_{clear} be the corresponding clear-sky estimate, the clear-sky index k^t is calculated as follows:

$$k^t = \frac{GHI^t}{GHI^t_{clear}} \tag{4.1}$$

where k^t nominally ranges from 0 to 1. However, its maximum value can be greater than one due to the cloud enhancement caused by diffuse sunlight. The estimated clear-sky value GHI^t_{clear} at time t is obtained from the regression curve that best fits the distribution of historical observations [10].

4.2.2 Assumptions of Consistency of Projected Images Across TSIs

The vertical height and horizontal expansion of clouds vary considerably among different types of clouds. Therefore, the projected size of a cloud observed by ground-based cameras depends strongly on the cloud type, the location of each camera, the field of view, and the solar zenith angle. With no loss of generality, we assume that the dimensions of the projected views of the cloud base in images from multiple TSIs are identical to ensure the suitability and mathematical correctness of the following calculations. In practice, our cloud tracking system does not enforce these strong assumptions; however, it is sufficiently robust to tolerate the differences between projected images while still being able to identify the same object in different images based on the values of the temporal and spatial correlations.

4.2.3 Consistency of Motion and Displacement Vectors

In this section, we assume that the motion vectors are consistent in spatial representation across all TSIs and that the displacement vector $\mathbf{d}^{i,j,t}$ is uniquely mapped to the cloud base height h^t. The pixel coordinates, (x_i, y_i), for TSI_i can be transformed into a coordinate system relative to the center of the camera (x_{ci}, y_{ci}). The azimuth and zenith angles (A_i and Z_i) at this point are represented

using the image coordinates as follows:

$$rx_i = x_i - x_{ci}\ ,\ ry_i = y_{ci} - y_i \tag{4.2}$$

$$Z_i = \arctan \frac{\sqrt{(rx_i^2 + ry_i^2)} \times \tan\theta_i}{R_i} \tag{4.3}$$

$$A_i = \arctan \frac{rx_i}{ry_i} \tag{4.4}$$

θ_i and R_i are pre-defined parameters in preprocessing to control FOV of the undistorted image for TSI_i. θ_i stands for the max zenith angle of FOV while R_i is the radius of visible scope at pixel level. Given the base height h, we can recover the 3-D spatial coordinates (sx_i, sy_i, sz_i) relative to TSI_i by

$$sx_i = h \times \tan Z_i \times \sin A_i \tag{4.5}$$

$$sy_i = h \times \tan Z_i \times \cos A_i \tag{4.6}$$

$$sz_i = h \tag{4.7}$$

Then, mapping from the image coordinates to the spatial coordinates can be expressed as follows:

$$sx_i = \frac{h \times tan\theta_i}{R_i} \times (x_i - x_{ci}) \tag{4.8}$$

$$sy_i = \frac{h \times tan\theta_i}{R_i} \times (y_{ci} - y_i) \tag{4.9}$$

If the geo-difference between TSI_j to TSI_i is defined as $(D_{i,j,x}, D_{i,j,y})$, then the same object in spatial coordinates system relative to TSI_j can be acquired

through the following equation:

$$sx_j = sx_i + D_{i,j,x} \ , \ sy_j = sy_i + D_{i,j,y} \ , \ sz_j = sz_i \tag{4.10}$$

Combine Equation 4.8, 4.9 and 4.10, we can derive image coordinates (x_j, y_j) to be:

$$x_j = \frac{R_j \times \tan\theta_i}{R_i \times \tan\theta_j} \times (x_i - x_{ci}) + \frac{R_j \times D_{i,j,x}}{h \times \tan\theta_j} + x_{cj} \tag{4.11}$$

$$y_j = y_{cj} - \frac{R_j \times \tan\theta_i}{R_i \times \tan\theta_j} \times (y_{ci} - y_i) - \frac{R_j \times D_{i,j,y}}{h \times \tan\theta_j} \tag{4.12}$$

Since the preprocessing of undistorting images uses the same scale and view range for all TSIs, from Equation 4.11 and 4.12, we derive the following:

$$\Delta x_i = \Delta x_j \ , \ \Delta y_i = \Delta y_j \ , \ when \ \theta_i = \theta_j \ , \ R_i = R_j \tag{4.13}$$

where Δx_i and Δy_i stand for the offset (distance) between pixel $\mathbf{c} = (x_i, y_i)$ and pixel $\mathbf{c}' = (x_i', y_i')$ along the x and y direction respectively in an image of TSI_i. Based on our assumptions and Equation 4.13, its projections on multiple TSIs have an identical size. Moreover, its pixel-wise motion is also the same since it stands for the same scale of movements on the image plane.

To retrieve the cloud base height, we define the mathematical term $\mathbf{d}^{i,j,t}$ as the pixel-wise displacement vector between two blocks in different TSI images (TSI_i and TSI_j). In detail, $\mathbf{d}^{i,j,t} = (d_x^{i,j,t}, d_y^{i,j,t})$ as the displacement vector in x and y directions in image at timestamp t. Based on the result in Equation 4.13, we acquire the height-to-displacement mapping as follows:

$$d_x^{i,j,t} = x_j - x_i = x_{cj} - x_{ci} + \frac{R_j D_{i,j,x}}{h^t \times \tan\theta_j} \tag{4.14}$$

$$d_y^{i,j,t} = y_j - y_i = y_{cj} - y_{ci} - \frac{R_j D_{i,j,y}}{h^t \times \tan\theta_j} \tag{4.15}$$

Therefore, the cloud base height, h^t, can be represented by the displacements in both the x and y directions respectively:

$$h^t = \frac{D_{i,j,x} R_j}{(d_{i,j,x} + x_{ci} - x_{cj}) \tan \theta_j} \tag{4.16}$$

$$h^t = \frac{D_{i,j,y} R_j}{(-d_{i,j,x} - y_{ci} + y_{cj}) \tan \theta_j} \tag{4.17}$$

Equation 4.16 and 4.17 should have the same output. Otherwise the displacement vector is invalid since it does not satisfy the spatial representation.

4.3 Cloud Detection and Block Generation

In order to separate clouds from clear sky pixels, a SVM-based classifier, as introduced in previous chapter, is adopted to divide all pixels within field of view into two categories: sky and cloud. Tracking clouds at a very small scale, for example at a pixel level could be sensitive to noise and the changes in luminance. Hence, we aggregate cloud pixels that generated in prior into cloud blocks. Each cloud block is used as a unit of cloud tracking. The block-matching algorithm, as shown in Huang's work [47], is to find the best match for a certain block so as to estimate the motion during this time period. The state-of-art block matching has been proved to have stable performance in cloud motion estimation if a proper block size was selected. Similarly in our approach, we restrict the range of block size to between 10 x 10 to 80 x 80. This guarantees that each cloud block has adequate textural variability and at the same time includes less unrelated pixels.

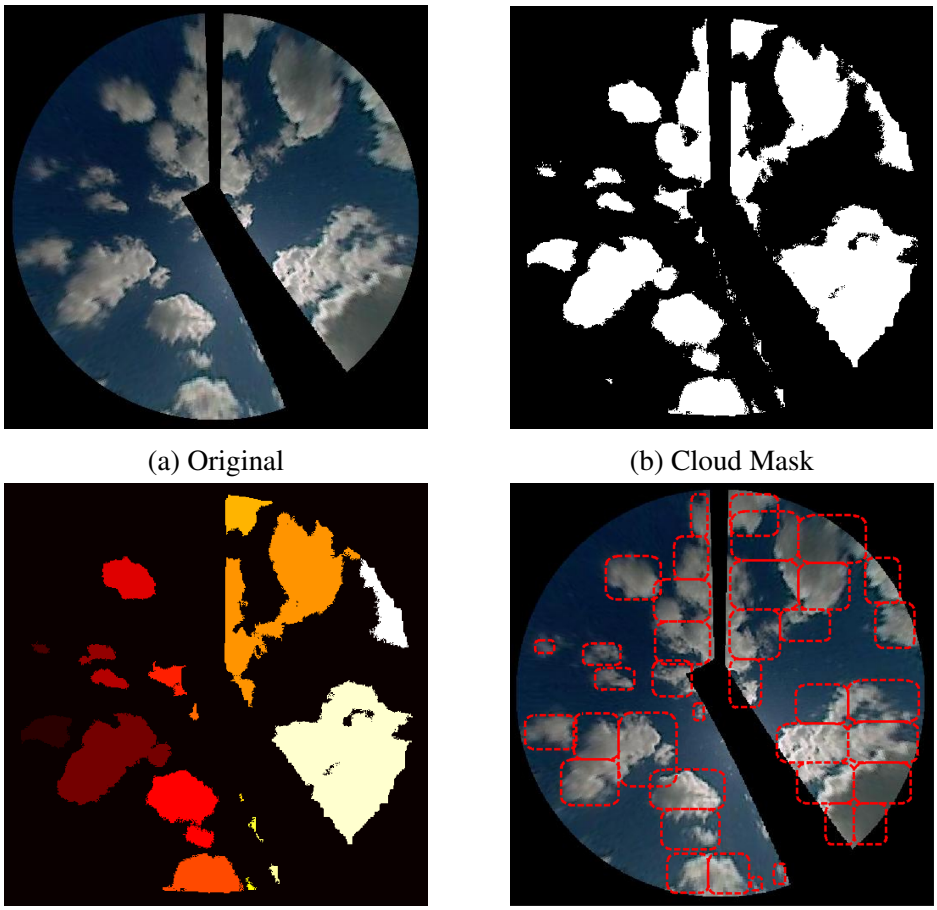
To aggregate cloud pixels into blocks of an appropriate size, we first apply the connected-component detection algorithm [129] to determine the regions that are distinct from clear sky. As shown in Figure 4-3c), each component marked with a given color has a high likelihood of belonging to the same

piece of cloud. For each component, we draw a minimum bounding rectangle box/block to enclose its cloud pixels. However, these bounding blocks may not be of a suitable size for cloud tracking because of the arbitrary size and shape of the connected components. Therefore, we apply the iterative rectangle split-and-merge algorithm to all components to ensure that the final cloud blocks have acceptable dimensions in accordance with the pre-defined range (i.e., 10x10 to 80x80). As shown in Figure 4-3d, the final output is a sequence of cloud blocks derived from connected components, each of which is treated as a unit region for calculating the temporal and spatial correlations of its internal cloud pixels.

4.4 Cloud Base Height and Motion

Using the generated cloud blocks, the basic concept of cloud tracking is to recover the 3D coordinates for each block and estimate their motions. Figure 4-4 shows an overview of three-TSI tracking. A piece of cloud at a certain level can be projected into/visualized as three different cloud blocks on the projection planes of the three difference TSIs. Because of the geometric difference among the three TSI devices on the ground, the pixel coordinates of each cloud block are distinct. Therefore, these cloud blocks can be combined to recover their 3D coordinates. Theoretically, a pair of TSIs should suffice to recover the cloud base height (CBH) using a stereography approach [138] [139] [140]. In our system, we utilize the redundant information from the three-TSI network to increase the robustness of cloud tracking.

To retrieve the CBH, we define the mathematical term $\mathbf{d}^{i,j,t}$ as the pixel-wise displacement vector between two blocks in different TSI images (TSI_i and TSI_j). If I represents the image matrix of an undistorted image and $\mathbf{c} = (x, y)$ is the center of a cloud block, then the displacement of $TSI_i \rightarrow TSI_j$ at time t is expressed as $\mathbf{d}^{i,j,t} = (d_x, d_y)$ and satisfies the following:



(a) Original (b) Cloud Mask

(c) Component Map (d) Cloud Blocks

Figure 4-3: Generating cloud blocks from a cloud mask. (a) Original image. (b) The cloud mask generated from the original image; (c) The output components detected by applying connected components detection are illustrated in different colors. Using a "split and merge" technique, the final cloud blocks are generated as shown in red in (d).

.

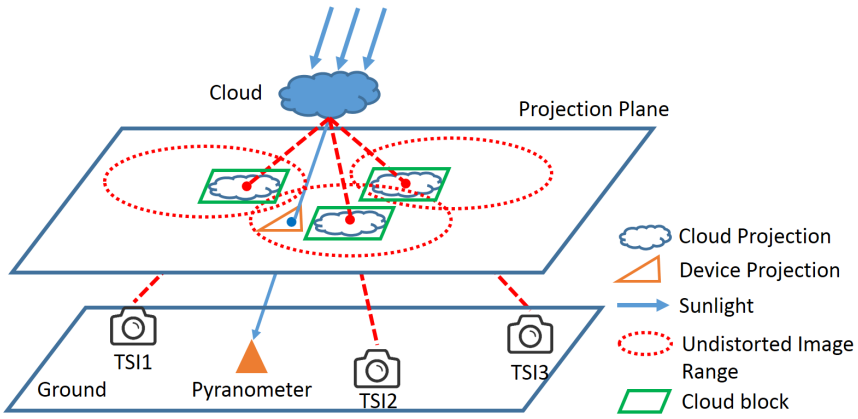


Figure 4-4: Overview of three-TSI tracking.

$$I_i^t(\mathbf{c}) = I_j^t(\mathbf{c} + \mathbf{d}^{i,j,t}) \tag{4.18}$$

If h^t denotes the CBH detected at time t, based on computational geometry, then there exists a unique mapping f that can be used to calculate h^t from the displacement vector (see Appendix 4.2.3):

$$h^t = f_{i,j}(\mathbf{d}^{i,j,t}) \tag{4.19}$$

Conversely, if h^t is calculated from one displacement vector, then all other displacement vectors can be calculated by mapping the derived 3D coordinates back to the projection plane. We define this procedure as a fixed series of transformations g, which are determined only by the locations of the TSI (see Appendix 4.2.3).

$$\mathbf{d}^{i,j,t} = g_{i,j}(h^t), i, j \in S = \{1,2,3\} \tag{4.20}$$

To track the cloud movement over time, we define another similar term, the motion vector, to describe the pixel-wise shift of a cloud block between consecutive frames. We adopted the traditional Optical Flow (OF) estimation [141] to define the motion vector from the time frame t to the next frame $t+1$ for TSI_i

as $\mathbf{v}^t = (v_x^t, v_y^t)$. Assuming consistent brightness, the motion vector satisfies the following equation:

$$I_i^t(\mathbf{c}) = I_i^{t+1}(\mathbf{c} + \mathbf{v}^t) = I_i^{t+1}(x + v_x^t, y + v_y^t) \tag{4.21}$$

Given the definition of the motion and displacement vectors, we simplify the cloud tracking task by adopting two assumptions: a). clouds exhibit only planar movement, without any vertical motion, and b). the velocity and direction remain constant within our forecasting time window. Moreover, as mentioned in Section 4.2.1, all undistorted images reside on the same projection plane, with the identical fields of view and spatial resolutions. Therefore, to ensure the uniformity of the TSI images and the consistency of the cloud movements without any loss of generality, we make several additional assumptions about our three-TSI system. First, a cloud block and its counterparts in other TSI images have the same dimensions, without scaling or shearing. Figure 4-4 shows that the cloud blocks obtained from different TSIs to represent the same piece of cloud must be identical in size on the same plane. Second, the mapping from the displacement vector to the CBH is independent of time because the mapping functions f and g are fixed and determined only by the locations of the TSIs (see Appendix 4.2.3). Third, a cloud block maintains a consistent size over a short time. Because a piece of cloud is assumed to exhibit only planar movement, all its cloud blocks residing on the projection plane must be of the same size at different timestamps. Finally, the fourth assumption is that a cloud block registered by one TSI and its counterparts at the other TSIs have the same motion vector. More importantly, the scale and direction of the motion are consistent, as based on the previous assumption (see Appendix 4.2.3). We note that our cloud tracking system does not enforce these strong assumptions in practice; however, it remains sufficiently robust to tolerate the differences between the projected images and still identify the same object from different images based on their

temporal and spatial correlations.

Based on these assumptions, we designed a novel tracking system utilizing all three TSI devices ($S = \{1,2,3\}$) for a series of consecutive timestamps ($T = \{t, t+1, t+2\}$). In total, nine images are used to extract two types of shift vectors at the pixel level: 1) the displacement vectors between TSI views to calculate CBH, and, 2) the motion vectors between consecutive frames from one TSI (planar motion). An intuitive example is illustrated in Figure 4-5. After identifying a cloud block segment in an image from TSI_i at time t, our goal is to find its "best" matches in the other eight images, and use these matches to explicitly calculate its motion and displacement vectors. On one hand, the motion vectors of the nine cloud blocks should be identical to $\mathbf{v}^t$ because they represent the same cloud with steady movements. Hence, we simplify the tracking problem by setting $\mathbf{v}^t = \mathbf{v}^{t+1}$ during block matching. Moreover, we assume that the cloud base height remains consistent within a short tracking window. As a result, the estimated CBH derived from the "best" matches must satisfy $h^{t+1} = h^t$, which is also essentially equivalent to $\mathbf{d}^{i,j,t+1} = \mathbf{d}^{i,j,t}$. To identify the best estimates of $\mathbf{v}^t$ and $\mathbf{d}^{i,j,t}$, the next step is to define a criterion to match multiple cloud blocks in different images.

4.5 Cloud Tracking Similarity Function

For this system, we have designed an algorithm to simultaneously incorporate both spatial and temporal correlations to improve the accuracy of the block-matching methodology. We proposed formulating the tracking problem as the maximization of a similarity function in which these two types of correlation are summed. In detail, this function consists of two components: 1) the sum of the similarity among different time frames for each TSI, and 2). the sum of similarity between any pair of TSIs at a given timestamp. Given a cloud block centered at $\mathbf{c}$ on TSI_1 at time t, the similarity function, φ can be calculated

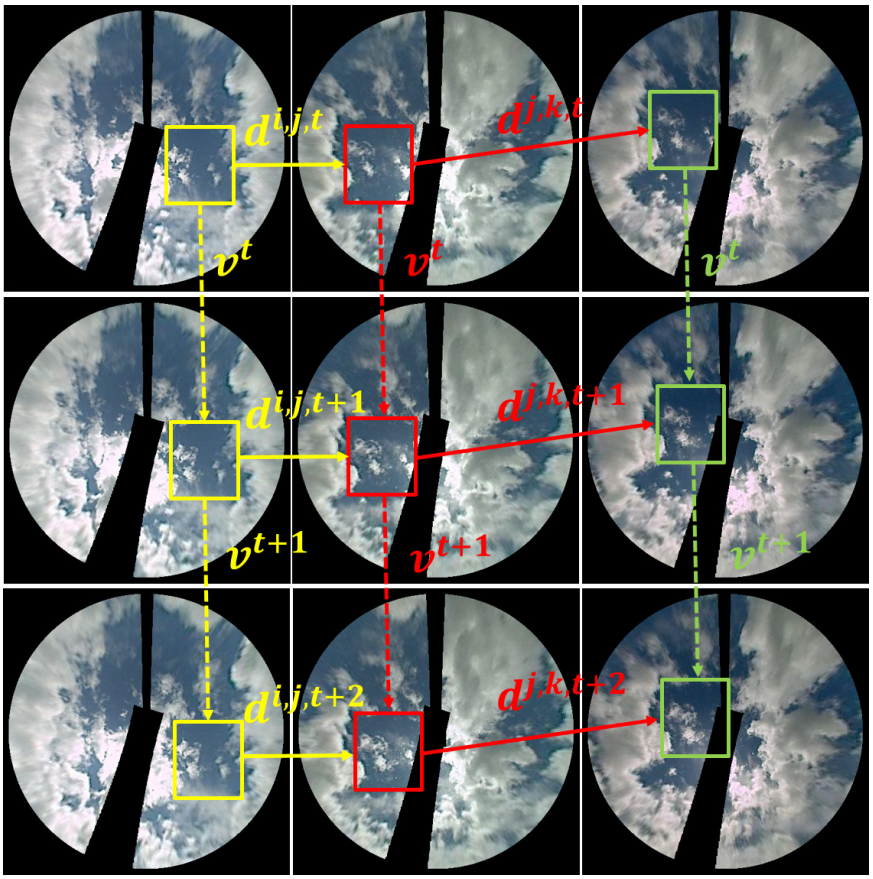


Figure 4-5: Matching cloud blocks in nine images. The cloud block of interest is indicated in yellow, red, and green boxes on the TSI_1, TSI_2, TSI_3 images, respectively. The movement of the cloud block that is detected between two consecutive frames is indicated by a dotted arrow and labeled as **v**. The displacement vector between a pair of TSIs at the same timestamp is represented by a solid arrow and labeled as **d**.

by summing all similarity values, and the motion vector and one displacement vector are denoted by $\mathbf{v}^t$ and $\mathbf{d}^{1,2,t}$ respectively. The formula are as follows:

$$\begin{aligned}\varphi(\mathbf{v}^t, \mathbf{d}^{1,2,t}, \mathbf{c}, t, S) = &\sum_{i,j \in S, i \neq j, t \in T} NCC(I_i^t, I_j^t, \mathbf{c}, \mathbf{d}^{i,j,t}) \\ &+ \sum_{i \in S, t' > t, t' \in T} NCC(I_i^t, I_i^{t'}, \mathbf{c}, (t'-t) * \mathbf{v}^t)\end{aligned} \tag{4.22}$$

$$\mathbf{d}^{i,j,t} = g_{i,j}(f_{1,2}(\mathbf{d}^{1,2,t})) \tag{4.23}$$

Here the two functions $f_{i,j}$ and $g_{i,j}$ can be obtained: $f_{i,j}$ is one of Equations 4.16 and 4.17 while $g_{i,j}$ is chosen from Equations 4.14 and 4.15. Therefore we conclude that given a pair of displacement vector $\mathbf{d}^{i,j,t}$, h^t is uniquely determined by $f_{i,j}$. More importantly, the mapping function series, f and g, are time-independent as introduced in Section 4.2.3.

Note that we include only the displacement vector from TSI_1 to TSI_2 in the similarity calculation. The displacement vectors between any pair of TSIs can be derived similarly using Equation 4.23. Here, *NCC* essentially represents the normalized cross correlation between two image blocks of the same size and dimensions, and it is chosen as the criterion for matching image blocks [129]. The formula for *NCC* is expressed as follows:

$$\begin{aligned}&NCC(I, J, \mathbf{c}, \mathbf{u}) = \\ &\quad \frac{1}{p} \sum_{\mathbf{k}=(i,j)} \frac{\big(I(\mathbf{c}+\mathbf{k}) - \mu(I, \mathbf{c})\big)\big(J(\mathbf{c}+\mathbf{u}+\mathbf{k}) - \mu(J, \mathbf{c}+\mathbf{u})\big)}{\sigma(I, \mathbf{c})\sigma(J, \mathbf{c}+\mathbf{u})}\end{aligned} \tag{4.24}$$

where I and J are the grayscale image matrices converted from the undistorted images. Each pixel value lies in the range $(0, 255)$. Furthermore, $\mathbf{u}$ is a motion or displacement vector. Assuming that the block size is (m,n), then $i \in \{-m/2, -m/2+1, ..., 0, ..., m/2\}$ and $j \in \{-n/2, -n/2+1, ..., 0, ..., n/2\}$ are the pixel indices of the cloud block, $p = m \times n$ is a normalization constant,

μ is a mean function, and σ is a standard deviation function of the image block. *NCC* is widely used in motion estimation because of its simplicity and robustness to noise and changes in intensity.

Given the definition of the similarity function φ, our goal is to search for a combination of ($\mathbf{d}^{1,2,t}$, $\mathbf{v}^t$) that optimizes its value. The most straightforward method is searching all combinations and finding the $(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t)$ that maximizes the φ score:

$$(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t) = \underset{\mathbf{d}^{1,2,t}, \mathbf{v}^t}{argmax} \ \varphi(\mathbf{v}^t, \mathbf{d}^{1,2,t}, \mathbf{c}, t, S) \tag{4.25}$$

However, the maximum of φ is only the correct result if a) the cloud block is located in the middle of the FOV and visible in all nine images; b) the image quality is sufficient to accurately display the cloud texture within the block; and, c) the cloud inside the block and its counterparts in the images from the other TSIs have similar shape and size, even from different angles of view. In general, several difficulties prevent us from attaining the optimal solution to maximize φ, as described below. In the previous work of [48], the block tracking near the image boundary was found to be inaccurate because of the loss of information. Only the blocks that are at least a certain distance from the boundary can possibly be matched using *NCC* value. Moreover, because cloud blocks have various sizes and textures, the tracking performance achieved by applying *NCC* may vary. In practice, even for cloud blocks that are distributed on the same layer and exhibit identical movement, their optimal solutions to Equation 4.25 which should be identical may be similar, but no identical. Another difficulty encountered in cloud-block tracking using TSI images is that some information is missing due to artifacts from the TSI. Blank pixels (marked as black in the TSI images) significantly impact block-matching since the maximization of *NCC* always favors blocks with lower loss of information. Furthermore, noisy pixels or variations in image brightness may cause instabilities in calculating

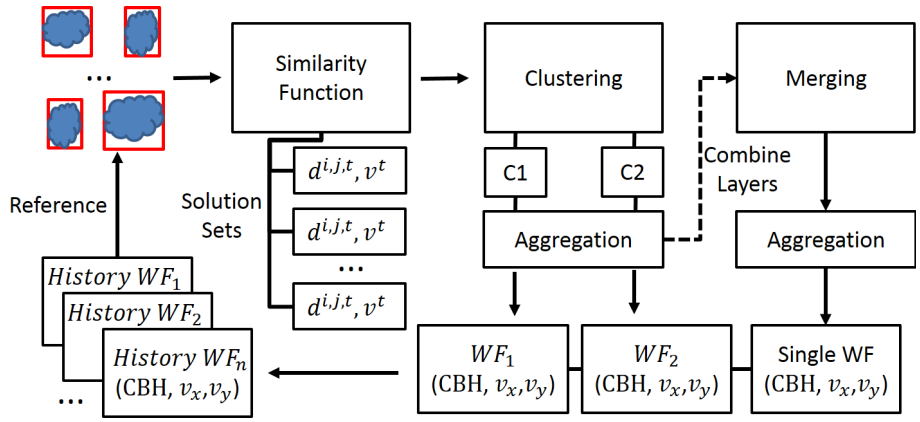


Figure 4-6: Overview of cloud-block tracking and the determination of multi-layer wind fields.

the *NCC* score and thereby influence the cloud-tracking performance based on Equation 4.25. Consequently, the maximum value of φ does not always guarantee the best matches among nine images. In practice, we use $(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t)$ only as a reference solution for determining the layer information.

As discussed in the previous section, static features are not effective for differentiating cloud pixels near the sunspot. We propose incorporating the dynamic information from cloud tracking to enhance cloud detection in the sunspot region. To mitigate the influence of false-positive pixels around the sunspot, we apply a pre-defined sunspot mask (window) and acquire the motion vector $\mathbf{v}_m^t$ by solving Equation 4.25 for each cloud block within this range. If the magnitude of $\mathbf{v}_m^t$ for a cloud block is close to 0, then in the subsequent cloud-layer determination step, we convert the relevant cloud pixels within this block into clear-sky pixels.

4.6 Multi-layer Detection and View Stitching

To locate the best match and find the optimal solution of the similarity function φ, we introduce clustering and multi-layer aggregation during cloud tracking to utilize all cloud blocks in the visible range. Instead of considering only the

maximum of φ, we propose tracking all possible $(\mathbf{d}^{1,2,t}, \mathbf{v}^t)$ combinations as potential solutions. In practice, we store a combination for a cloud block only if each *NCC* score in the φ calculation is above a certain threshold (here, we use 0.5). Therefore, a single block may have multiple motion and displacement vectors. Our goal is to obtain $\mathbf{L}$, a collection of multiple potential $(\mathbf{d}^{1,2,t}, \mathbf{v}^t)$ combinations detected for all cloud blocks:

$$\mathbf{L} = \{(\mathbf{d}^{1,2,t}, \mathbf{v}^t) | NCC^* \geq 0.5, \mathbf{c} \in C\} \tag{4.26}$$

Note that NCC^* refers to any *NCC* calculation listed in the φ equation. C denotes the set of all cloud blocks detected in the cloud detection pipeline.

Based on $\mathbf{L}$, we apply the k-means clustering [142] technique to generate layers of clouds. In our system, two layers at most are considered and aggregated. Hence, binary clustering is used to split $\mathbf{L}$ into two categories. If we use the clustering result to represent cloud layers, then the centroid of a cluster, $(\mathbf{d}^{1,2,t}, \mathbf{v}^t)$, represents the primary height and motion of this layer. Thereafter, we can group all cloud blocks into two layers/clusters based on the Euclidean distance between the reference solution $(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t)$ and the centroids. The entire cloud block set C is then divided into two layers, C_1 and C_2, each of which contains multiple cloud blocks that should have similar motions and heights. To generate more accurate and robust information concerning cloud layers, we assume that cloud blocks on the same layer possess only one major planar motion vector and one CBH. Similar to the form of the centroid in clustering, we define the wind field (WF), $(\mathbf{d}^{1,2,t}, \mathbf{v}^t)$, as the unique combination of the displacement and motion vector corresponding to a given cloud layer. We calculate this combination of vectors by maximizing the summation of φ over all cloud blocks on the same layer:

$$WF_i = \underset{\mathbf{d}^{1,2,t}, \mathbf{v}^t}{argmax} \sum_{\mathbf{c} \in C_i} \varphi(\mathbf{v}^t, \mathbf{d}^{1,2,t}, \mathbf{c}, t, S) \tag{4.27}$$

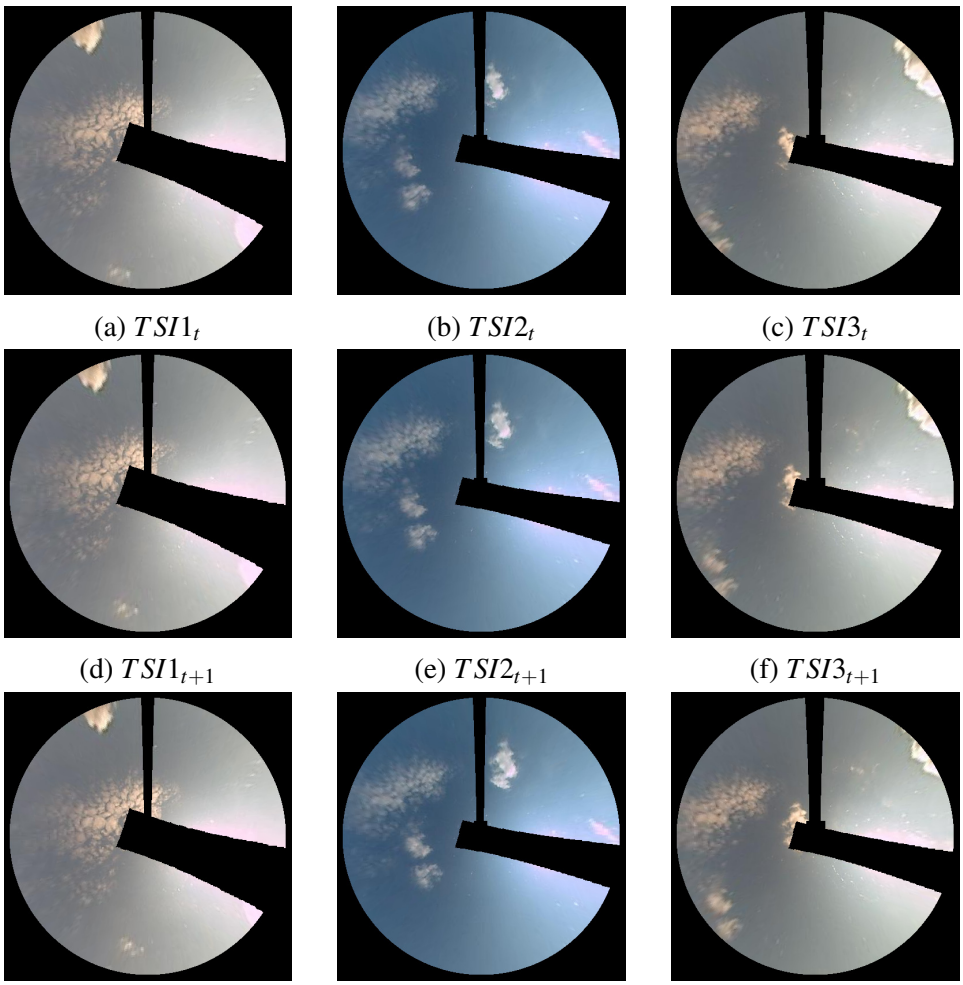
If two wind fields extracted from two clusters are sufficiently similar to each other, then we treat them as a single cloud layer in which $\mathbf{c}$ can be drawn from the entire cloud block set C. Similarly, the single wind field can be expressed as follows:

$$WF_{single} = \underset{\mathbf{d}^{1,2,t},\mathbf{v}^t}{argmax} \sum_{\mathbf{c}\in C} \varphi(\mathbf{v}^t, \mathbf{d}^{1,2,t}, \mathbf{c}, t, S) \tag{4.28}$$

When the Euclidean distance between two motion vectors is less than two pixels, or their height difference is less than 500 meters, we consider these two wind fields to belong to a single layer. To further improve tracking performance and reduce computational complexity, all generated wind fields are stored as historical layers. When searching for a reasonable solution set, this historical reference is used to refine the possible range of motion vectors and height levels. This approach accelerates the search procedure and increases the system's robustness in coping with noise.

An example with two (layers of) wind fields detected is shown in Figures 4-7 and 4-8. Nine images in three consecutive frames from timestamp t to $t+2$ are used to extract potential solutions. In Figure 4-8, the displacement vectors of $TSI_2 \rightarrow TSI_3$ and $TSI_2 \rightarrow TSI_1$ are shown in blue and green respectively, whereas the cloud motions are shown in red. Figures 4-8a-4-8c confirm that to maximize φ for each individual block does not always guarantee finding the best matches. We observe that cloud blocks near the shadow band and supporting arm (black area) exhibit obvious errors due to the information loss caused by blank pixels. However, after clustering and aggregating, the cloud blocks can be categorized into two layers, $WF_1 = (1201m, 10px, -6px)$ and $WF_2 = (4184m, 7px, -2px)$. As shown in Figure 4-8d- 4-8f, these two layers are indicated by red and green boxes, respectively, and offer more stable estimations of displacement and motion than do the individual cloud blocks.

We also present another example of a single WF detected from the nine in-



(a) $TSI1_t$ (b) $TSI2_t$ (c) $TSI3_t$

(d) $TSI1_{t+1}$ (e) $TSI2_{t+1}$ (f) $TSI3_{t+1}$

(g) $TSI1_{t+2}$ (h) $TSI2_{t+2}$ (i) $TSI3_{t+2}$

Figure 4-7: Example of cloud tracking on nine images.

put images shown in Figure 4-5. In Figures 4-9a- 4-9c, several combinations of height and motion that achieve the maximum value of φ are deemed outliers and marked with green rectangles. Running the clustering algorithm identified two wind fields: $WF_1 = (4357m, 8px, 8px)$, and $WF_2 = (2098m, 8px, 9px)$. However, these two layers are very similar to each other given their similar motion vectors. Thus, we aggregated all blocks into one layer, as indicated by the red boxes in Figure 4-9d- 4-9f. The output WF_{single} is then represented as a single tuple $(2130m, 8px, 9px)$ which is less sensitive to the boundary or noisy cases.

After determining the cloud layers, multiple TSI views can be stitched to-

gether block by block. We select TSI_2 as the origin/center of this combined view because it is located in the middle. The pixels of a cloud block from TSI_1 and TSI_3 can then be mapped to TSI_2's coordinate system through pixel-wise shifts of the forms $TSI_1 \rightarrow TSI_2$ and $TSI_3 \rightarrow TSI_2$ which are identical to the extracted displacement vectors $\mathbf{d}^{\mathbf{1,2,t}}$ and $\mathbf{d}^{\mathbf{1,3,t}}$. Thus, given the estimated heights of the cloud layers, all cloud blocks from TSI_1 and TSI_3 are placed and stitched into the aggregated view from all three TSIs. An example of such a stitched view wherein all blocks belong to a single layer is shown in Figure 4-9g. For a case with more than one layer, the stitched view is generated by stitching layers one by one, i.e., from a higher altitude layer to a lower one, because low-altitude clouds can overlay those at higher altitudes. Consequently, the output view should be similar to Figure 4-8g. The black areas where gaps still exist are marked as blank, and we fill them in with the default sky color for visualization. The default value is calculated by averaging all clear-sky pixels across multiple TSI views.

4.7 Short-term Solar Forecast

In this section, we describe the methods for extracting image features and the different irradiance models used to forecast solar irradiance. First, given the locations of the 25 pyranometers, we need to identify the relevant pixels on the TSI images that are correlated with the surface irradiance fluctuations at these locations. Since the clouds located between the sun and the solar panels are the primary cause of ramps in irradiance, the basic concept of irradiance forecasting is to predict whether clouds will block the sunlight at a specified time. As shown in Figure 4-4, the device's projection on the sky image corresponds to the intersection of the projected plane (i.e., the image plane) with a line drawn from the sun to the pyranometer. If a cloud is located in the path of the sunlight traveling to the pyranometer, then its projected position on the image plane must overlap

with the device's projection. Furthermore, we define the corresponding pixel at the projected position in the stitched view of the three TSIs as a "sun-blocking" pixel. Because this pixel is correlated with the amount of direct sunlight cast on a pyranometer, our goal is then to extract the image features of this pixel which will subsequently serve as the input to the irradiance model. As shown in Figure 4-4, the position of the sun-blocking pixel depends on the angle of the Sun, the geolocation of the pyranometer, and the height of the cloud that is blocking the direct sunlight. Using these inputs, we can easily calculate the sun-blocking pixel for each of the 25 pyranometers, as shown in Figure 4-10a and 4-10b.

To predict which pixels will become sun-blocking pixels in the future, we apply a backtracking method to the current stitched view. In this processing step, the backtracking is guided by the motion vectors detected at previous timestamps: given that the cloud motion in a particular layer is $\mathbf{v^t}$ where t is the current timestamp, the pixel $\hat{\mathbf{c}}^t$ that will potentially become a sun-blocking pixel $\mathbf{c}^{t+N}$ in the Nth future time frame (at time $t+N$) is then calculated by moving $\mathbf{c}^{t+N}$ in the direction opposite to the motion vector $\hat{\mathbf{c}}^t \leftarrow \mathbf{c}^{t+N} - \mathbf{v^t} \times N$. Here, we assume that the motion vector remains unchanged between t and $t+N$ and that the pixel at $\hat{\mathbf{c}}^t$ will become $\mathbf{c}^{t+N}$ and block the sunlight after N time frames, provided that it contains a cloud at that time. If multiple layers of clouds move to this pixel location, then the lower-altitude cloud pixel is preferred because it will block the higher-altitude one.

As expected, the effectiveness identification of the sun-blocking pixel requires an accurate estimation of the motion vectors of each layer. Any error in this information will further accumulate in subsequent steps. To mitigate this effect, we have devised two strategies to be implemented in the feature extraction step. First, instead of focusing on a single sun-blocking pixel, we considered the 7x7 sun-blocking window surrounding this pixel. This enables us to reduce the risk of false prediction, and more importantly, to include more features from the neighboring pixels. Second, we use 23 significant features to describe the spec-

tral variation inside the window at the current time, t and the predicted time, $t+N$. Among these features, nine are the average, minimum, and maximum values of the RGB channels of the window surrounding $\mathbf{c}^t$, which describe the spectral properties of the current observation. Similarly, we choose nine additional features from the sub-blocking window centered at $\hat{\mathbf{c}}^t$, which represent the estimated properties. The remaining five features, are the RBR of $\mathbf{c}^t$ (RBR^t) and $\hat{\mathbf{c}}^t$ ($R\hat{B}R^t$), the cloud fraction at time t, and the ground-truth irradiance values at the current time t (k^t) and one-minute earlier, at $t-6$ (k^{t-6}). The RBR at both timestamps are used because this quantity shows a noticeable spectral difference between cloud and sky [27]. The cloud fraction represents the overall cloud conditions. The two remaining features, the ground-truth irradiance values at t and $t-6$, enhance the performance for a short forecasting window because they incorporate persistent observations. Here, the forecasting problem is formulated as $\hat{k}^{t+N} = f(\mathbf{x}^t)$, where $\hat{k}^{t+N}$ represents the predicted irradiance at $t+N$ and $\mathbf{x}^t$ is the vector of the 23 extracted features.

To examine the predictive capabilities of the selected features, we explore four different irradiance models that use a subset or all of these features to generate a regular linear regression or more complicated non-linear relationships $f(\mathbf{x}^t)$: 1) linear RBR delta, 2) an ordinary linear regression model, 3) Support Vector Regression (SVR) [143] based on a linear kernel, and 4) SVR with a non-linear kernel. For comparison, we use the persistent model as the baseline model, i.e., the **radiation shift**, $RShift$, which directly uses the current observation as the forecasting result. In addition, we introduce one regression model that depends only on multiple irradiance values and is used to further validate image features for prediction capability in our comparison studies.

$$\hat{k}^{t+N} = k^t \tag{4.29}$$

First, we simply extend the persistent model by incorporating a linear regu-

larization term based on the information from the sun-blocking pixel. Our preliminary study of motion vectors [48] indicates that the RBR of a sun-blocking pixel is a useful indicator of the cloud transmittance at this pixel. The **linear RBR delta** model, denoted by $linear_{\delta}$, is as follows:

$$\hat{k}^{t+N} = k^t + C \cdot (R\hat{B}R^t - RBR^t) \tag{4.30}$$

where C is a negative coefficient. We then generalize the model $linear_{\delta}$ to an **ordinary linear regression model** ($linear_{all}$) over all attributes in $\mathbf{x}_t$:

$$\hat{k}^{t+N} = \mathbf{w} \cdot \mathbf{x}^t + b \tag{4.31}$$

where $\mathbf{w}$ is the weight vector, and b is the intercept. However, the ordinary model is sensitive to noise or outliers and suffers from the overfitting problem. To overcome these shortcomings, we apply the **Support Vector Regression** (SVR) [144], an extension of the SVM approach, for regression. Given a linear kernel, viz, SVR_{linear}, the radiation estimation still follows the ordinary form given in Equation 4.31, but the $\mathbf{w}, b$ pair is obtained by solving the classic SVR optimization problem:

$$\min_{\mathbf{w},b,\xi,\xi^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{4.32}$$

subject to

$$\begin{aligned} &(\mathbf{w} \cdot \mathbf{x}^t + b) - k^{t+N} \leq \varepsilon + \xi_t, \xi_t \geq 0, \forall t \\ &(\mathbf{w} \cdot \mathbf{x}^t + b) - k^{t+N} \geq -\varepsilon - \xi_t^*, \xi_t^* \geq 0, \forall t \end{aligned} \tag{4.33}$$

where ε is the regression margin, ξ_t and ξ_t^* are the slack variables, and C is a regularization parameter. By incorporating different kernels, we can extend the SVR to model a non-linear relationship by projecting the current attributes into high-dimensional transformed spaces, for example, Fourier space. To this end, we use the Radial Basis Function (RBF) kernel, SVR_{rbf}, with σ as the RBF

smoothing parameter:

$$\begin{aligned} \hat{k}^{t+N} &= \kappa(\mathbf{w}', \mathbf{x}^t) + b \\ &= e^{-\frac{\|\mathbf{w}' - \mathbf{x}^t\|^2}{2\sigma^2}} + b \end{aligned} \tag{4.34}$$

For both the linear and non-linear SVR models, certain parameters, such as, the regularization weight C, the error tolerance range ε and the RBF smoothness factor σ, must be tuned to achieve the optimal performance. For our experiment, we selected the optimal parameters from within certain ranges. Here, C is taken from $C = [-1024, 1024]$, ε from $\varepsilon = [0.001, 0.6]$, and σ from $\sigma = [10^{-4}, 10^{4}]$. The average values of the 25 solar sensors for SVR_{linear} are set as follows: $C_{avr} = 35, \varepsilon_{avr} = 0.01$. In the case of SVR_{rbf}, $C_{avr} = 19$, $\varepsilon_{avr} = 0.003$, and $\sigma_{avr} = 0.2291$. We trained the $linear_{all}$ and $linear_{\delta}$ models by minimizing their least-square errors. The persistent model $RShift$ requires no training.

4.8 Experiments

4.8.1 Experimental Dataset

We chose the period from May 13, 2013, to June 03, 2013, encompassing various weather conditions, to evaluate the performance of 1-min- to 15-min-ahead 3D cloud tracking and irradiance forecasting. To guarantee consistent cloud visibility and zenith angle, we filtered out records whose timestamps are not between 9:00AM and 16:00PM Eastern Standard Time (EST). Our experimental dataset contains 9963x3 images from three TSIs that collect data simultaneously and the synchronized 9963x25 GHI records from 25 ground-based pyranometers. Furthermore, we divided the full dataset into four categories based on weather and cloud conditions: single-layer clouds (D_s), multi-layer clouds (D_m), overcast or extremely cloudy (D_o), and a mixture thereof (D_{mix}). D_s corresponds to a typical low-altitude cloud that is commonly observed on the East Coast of

the United States. This type of low cloud typically appears in a single layer and remains in the field of view of a TSI for several minutes at most. Given its quick movement in and out of the FOV of a TSI, D_s is an ideal dataset for demonstrating the capability of the new tracking system to capture rapid cloud motion and evolution. D_m is a collection of cases wherein multiple layers of clouds were observed within the FOV of all TSIs. Hence, the tracking and forecasting performance with respect to D_m is expected to reflect the ability of the system to categorize and track multiple wind fields. D_o corresponds to cloudy and overcast cases. Based on our observations, the textural patterns of the images acquired in extremely cloudy conditions are not obvious and are hence difficult to utilize for block-matching and the extraction of image features. Therefore, we designed D_o to demonstrate the sensitivity of this new tracking system when applied to TSI images with less obvious patterns. D_{mix} contains the remaining cases, including sunny, partially cloudy, and thin-layer conditions. The cloud tracking and irradiance forecasting performance with respect to D_{mix} reflects the system's average performance for a mixture of cloud conditions. Table 4.1 describes the details of all subsets.

4.8.2 Evaluation Metrics and Validation Method

As discussed in Section 4.7, the prediction of the sun-blocking pixels for the 25 pyranometers is performed based on cloud movements in a particular cloud layer. When a cloud moves too rapidly, or the forecasting interval is too long, one or more positions may be outside of the FOV of our TSI. In this case, we cannot extract any features for modeling and predicting irradiance. To quantify the tracking capability for sun-blocking pixels within our stitched view, we define a metric named the Successful Tracking Index (STI), i.e., the percentage of data instances in the designated dataset for which all 25 sun-blocking pixels can be included in the FOV of the stitched image for a particular forecasting

horizon. The STI is calculated as follows:

$$STI = \frac{N - N_{oof}}{N} \tag{4.35}$$

Here, N is the total size of the dataset, while N_{oof} represents the number of records for which one or more sun-blocking pixels lie outside the field of view.

We also use the mean absolute error (MAE) and root-mean-square-error (RMSE) as evaluation metrics. In our experiment, the MAE measures the average accuracy of cloud tracking, whereas the RMSE assigns greater penalties to large errors, such as falsely estimation of a cloud's presence. To avoid bias and control the over-fitting problem, we introduce the cross-validation technique [120] into the modeling and evaluate the forecasting performance across all 25 stations. In the cross-validation, the original dataset is evenly divided into several independent subsets, and the average performance of the predictive model measured across these subsets. In this experiment, five-fold cross-validation ($N_{cv} = 5$) is applied, the five folds (four for training and one for testing) randomly generated. Additionally, since 25 simultaneous, location-dependent ground measurements are available ($s = 1, 2, ...25$), we can verify a forecast by comparing it with the observations from a different measurement station, s, at time t. Hence, the final error metrics can be formulated as the average performance of all five-fold tests across all 25 stations as follows:

$$MAE = \frac{1}{N_{cv}} \sum_{i=1}^{N_{cv}} \frac{1}{N_i} \sum_{s,t} \left| k_s^t - \hat{k}_s^t \right| \tag{4.36}$$

$$RMSE = \frac{1}{N_{cv}} \sum_{i=1}^{N_{cv}} \sqrt{\frac{1}{N_i} \sum_{s,t} (k_s^t - \hat{k}_s^t)^2} \tag{4.37}$$

Here, N_i is the size of the test fold, i, generated in the cross-validation. The MAE and RMSE score are calculated based on the normalized GHI values.

To evaluate the effectiveness of the image features extracted from our multi-

layer cloud detection and tracking system, we created a new reference model, SVR_k, for comparison. This model also uses SVR with only radiation observations as inputs and excludes all image-based features, in contrast to our proposed forecast model SVR_{rbf}. In greater detail, SVR_{rbf} and SVR_k share the same radial basis function as their SVR kernel and use identical parameter settings for both training and testing. The difference is that SVR_k uses only the l (here, we choose $l = 6$) most recent radiation values, i.e., $k^{t-5}, k^{t-4}...k^t$, as inputs. We apply both SVR_{rbf} and SVR_k to generate real-time irradiance predictions to validate the effectiveness of the image features in producing short-term solar forecasts.

4.8.3 Model Performance

Figure 4-12 shows the 1-15 minute STI scores of the entire dataset (bars) and the four subsets (lines). We observe that the STI scores for D_s and D_m decrease dramatically beginning with the nine-minute forecast, whereas the performance for the other two subsets remain relatively stable between one and fifteen minutes. Consequently, the success ratio for the entire dataset, *overall*, also decreases as the time horizon increases. The STI decreases for longer forecast horizons (longer than ten minutes) because of fast-moving clouds. According to our observations from the TSI images, on the East Coast of United States, these clouds are mostly distributed in a single layer with a cloud base height below 3000 meters. They often have a high velocity, as detected at the pixel level, and exhibit rapid formation/dissipation within a 10-minute window. Thus, given the limited visible range and TSI resolution, 3D cloud tracking can capture the majority of low-layer clouds for only up to nine minutes. Beginning at the 10-minute horizon, low clouds are highly likely to move out of the field of our stitched view. This cloud property, combined with the physical limitations of the tracking TSIs is consistent with the significant decrease in the STI of D_s at the nine-minute horizon, reaching almost 0 at the ten-minute horizon. Similarly, the STI of D_m

decreases after nine minutes. However, since more than one layer is present in D_m, the tracking results benefit from partial estimations from the higher layers which tend to be more stable and have slower pixel-wise motion vectors. For D_o and D_{mix}, most cases correspond to high-layer clouds or mixed conditions and are therefore suitable for ten-minute forecasts and beyond.

We evaluated the irradiance forecasting performance over the entire dataset based on the metrics of the MAE and RMSE scores, and the results are presented in Tables 4.2 and 4.3. We excluded the out-of-FOV data points and only trained and tested the irradiance models based on the remaining available data subset. During model training, we discovered that when STI is low, the models tend to overfit the data. One reason for this behavior is that an excess of out-of-FOV records leads to a lack of observations in the training folds. For instance, if we train the model using D_s for forecast horizons longer than ten minutes, we barely have enough training records to generate the forecasting model. Another reason is that we may introduce bias into the forecasting models. Since rapid-changing cases, such as those with low-altitude clouds, are excluded for long forecasting horizons, the forecasting models will place more weight on the "easy" cases, such as those corresponding to sunny and overcast conditions. Hence, if too many records in an experimental dataset are out of the FOV, or the STI value is below a certain threshold (here, 60%), we mark the result with an asterisk to indicate a partial forecast and a potential overfitting problem. When all records are out of the FOV ($STI = 0$), we denote this scenario with '-', indicating that no forecasting result is available (see Table 4.3).

In Figures 4-11a and 4-11b, we evaluate the effectiveness of the four forecasting models by comparing them with the persistent model for one- to fifteen-minute forecasts. To ensure that the sun-blocking pixels are contained within the stitched view for the majority of the training and test cases over the full forecast horizons, we use the experimental dataset D_{mix} which has the most stable STI values (Figure 4-12). Figures 4-11a and 4-11b show that $linear_\delta$ consistently

acquires less large forecast errors in comparison with the persistent model (as measured by RMSE), but exhibits the worst performance in terms of average accuracy of irradiance forecast (as measured by MAE), confirming that tracking a single sun-blocking pixel leads to a high risk of deviating from its real position and thereby falsely predicting the presence of clouds. By virtue of our proposed feature extraction in the sun-blocking window, this problem is mitigated in the other three models which consistently provide better results in both metrics. In comparison to $linear_{all}$ which was trained using least-square errors, the linear SVR approach, SVR_{linear}, tends to be more robust in terms of average errors (MAE); however, it is very sensitive to large errors (RMSE). The proposed non-linear model, SVR_{rbf}, effectively addresses this problem by mapping non-linear relations to linear ones in a higher dimensional space. As is evident from these two figures, SVR_{rbf} effectively reduces the occurrence of large errors and outperforms the other four models in terms of the RMSE and MAE metrics.

We also analyzed the performance of SVR_{rbf} on the entire dataset and compared it with $RShift$ across all 25 ground measurements. Figure 4-13 shows the percentage of reduction in MAE achieved using SVR_{rbf}. The blue shaded regions represent the upper and lower improvement bounds for all 25 pyranometers, whereas the mean improvements are plotted as blue dots with standard deviation bars. We observe that beyond a nine-minute horizon, the uncertainty in performance improvement, as indicated by the upper and lower bounds, increases with the increase in forecasting horizon. This is expected because many data points are out-of-FOV for the D_s and D_m subsets, which affects the reliability beyond the nine-minute horizon. Moreover, several successful cases that remain stable even beyond the nine-minute forecast are ”easy” ones, such as sunny and overcast conditions. Therefore, the persistent model $RShift$ can take advantage of these cases to minimize error and is thus difficult to outperform. In the same plot, we observe that both the difference between the upper and lower bounds and the standard deviation increase as the time horizon increases. We

observe that despite the expected uncertainties arising for a long-term forecast, the SVR_{rbf} model is nevertheless significantly superior to the persistent model, achieving at least a 26% improvement.

To apply this study to real-time forecasting, we averaged the SVR_{rbf} parameters over the entire dataset and constructed the radiation-only model SVR_k to generate 5-, 10-, and 15-minute forecasts for May 14, 2013. Figure 4-14 compares the forecasting results of SVR_{rbf} and SVR_k with the real measurements from the deployed pyranometers. In the individual figures, the normalized GHI predictions are converted back to real values using Equation 4.1. To be consistent with the irradiance plots in Figure 4-14, the root-mean-square errors (RMSE) in the caption are calculated directly from ground-truth measurements (real GHI values) without normalization. The gray/dark areas in these figures represent gaps with no prediction which include the cases corresponding to the low zenith angles during the early morning and late afternoon as well as the periods in which sun-blocking pixels were out of the FOV. The results show that the five-minute forecasts generated using SVR_{rbf} achieve good accuracy and capture most radiation ramps. Meanwhile, the number of detected ramps decreases for ten-minute and fifteen-minute forecasts because of the instability of the motion vectors and the occurrence of clouds that may reside outside the field of the stitched view. Moreover, because SVR_k only relies on radiation features, we observe that it exhibits a behavior similar to that of the persistent model, often failing to detect radiation fluctuations and generating false alarms based on previous irradiance trends. For a longer forecasting horizon, such as 10 or 15 minutes, the forecasting accuracy of SVR_k decreases rapidly, and the model cannot faithfully predict irradiance ramp events. Compared with SVR_k, SVR_{rbf} incorporates multiple features derived from predicted cloud movements and sky images and consequently, introduces fewer forecasting errors and captures more ramp events.

4.8.4 Experiment Findings and Discussion

Early experiments [48] on cloud tracking and short-term solar forecasting demonstrated that our early system based on a single TSI worked well for prediction of less than two minutes, quickly deteriorated when the prediction time horizon increased beyond two minutes, and eventually failed to outperform the persistent model. In contrast, our new forecast system using multiple TSIs not only enlarges its FOV for cloud observation, but also enriches cloud tracking with new critical features, a.k.a., cloud base height and motion layers. Compared with the single-TSI approach [48], our multi-TSI method is able to reduce prediction errors by more than 30% for one and two-minutes horizons in the same geographical location. More importantly, multi-TSI approach effectively expands the forecast horizon for fast-moving and low-altitude clouds and provides accurate fifteen-minute prediction for the majority of cloud conditions in the East Coast of Unite States where clouds tend to be active and volatile.

Furthermore, our experiment showed that the errors and uncertainty of motion estimation and prediction using TSI images accumulate over time and eventually result in a large deviation from real observations. This is the reason why many TSI-based approaches can not outperform a naive persistent model. We did not continue our comparison between the single-TSI approach and multiple-TSI approaches beyond two minutes because the single-TSI approach did not work for the complex cloud environment in the East Coast within this prediction regime. With the robust spatio-temporal correlations over nine frames, multi-TSI approach mitigates these issues during motion tracking and minimizes tracking errors via a multi-layer clustering that is possible only with 3-D spatial information. Based on thorough experimental evaluations, we confirmed that multi-TSI network indeed demonstrates superior stability and robustness in short-term solar irradiance forecast. Moreover, because our integration framework is modularized and cost-efficient, it was already deployed in a large utility-

scale solar farm, e.g., LISF, to meet the requirements of real-time data processing and short-term solar forecasting.

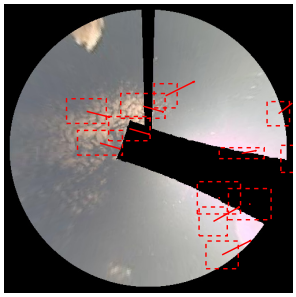
4.9 Summary

In this paper, we proposed a novel 3D cloud detection and tracking system using computational geometry and machine learning techniques. By taking advantage of redundant information from the overlapping views of multiple cameras, this system can not only identify clouds at the pixel level, but also determine their base heights and the wind fields of multiple cloud layers. This information can be used to stitch multiple TSI views together to generate an expanded cloud view that is larger than that obtained from a single camera, thereby increasing the forecast horizon. More importantly, based on the predicted layers and the stitched view, we investigated the use of image-based features and irradiance models to effectively capture short-term fluctuations in solar irradiance.

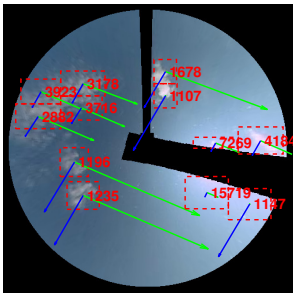
To verify the effectiveness of our tracking system, we organized our experimental data into four categories based on cloud conditions and tested the system over 25 ground sites equipped with pyranometers. The results showed that our new cloud tracking system provided reliable fifteen-minute forecasts for all 25 ground stations under most cloud conditions. Even for low-altitude and fast-moving clouds, this system can obtain reasonable estimates for up to nine minutes ahead. Furthermore, in our cloud tracking system, all four proposed irradiance models considerably reduced the prediction errors for forecasts of up to fifteen minutes ahead. Compared with the persistent model, both linear and non-linear models based on the extracted sky-image features significantly offered significantly improved accuracy with respect to the MAE and RMSE metrics.

In the future, we will explore new techniques for overcoming limitation in the FOVs of observations and introduce more useful features into the forecasting

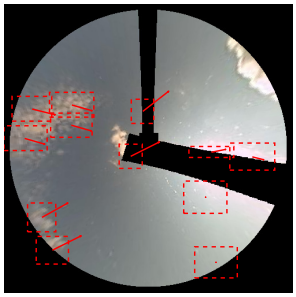
models. Although the stereo stitching of sky images from multiple TSI sources enlarges the visible range, it can only expand the forecasting range to a certain degree. A possible solution to this problem is to explore the statistical properties that can be derived from feature information in the cloud detection and tracking steps and to perform statistical estimations of the out-of-FOV cases. Moreover, our experiment demonstrated that individual image-scale features are prone to error accumulation over time. We will explore additional features from sky images and other sources, such as, LIDAR, temperature, and past predictions from the model itself, to improve its robustness and reduce forecasting errors.



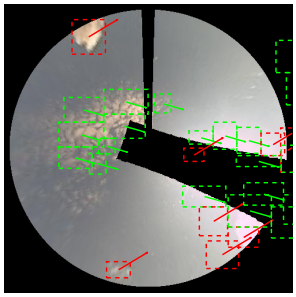
(a) Max. φ at $TSI1_t$



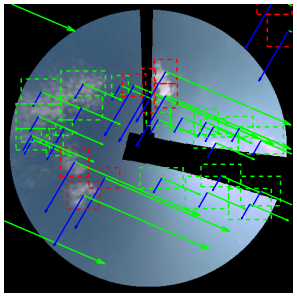
(b) Max. φ at $TSI2_t$



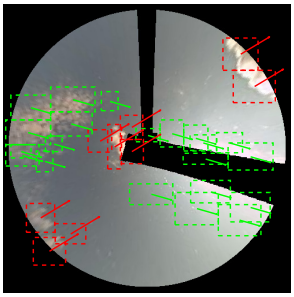
(c) Max. φ at $TSI3_t$



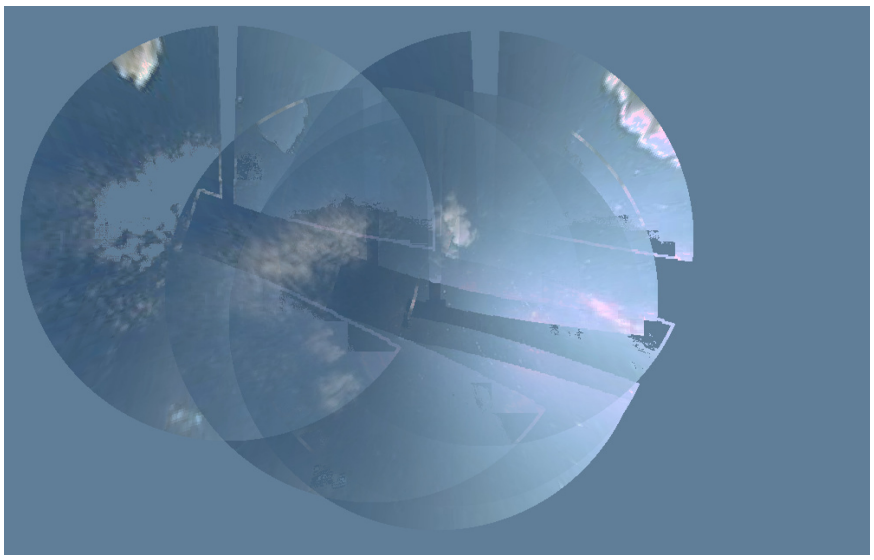
(d) Aggregation at $TSI1_t$



(e) Aggregation at $TSI2_t$

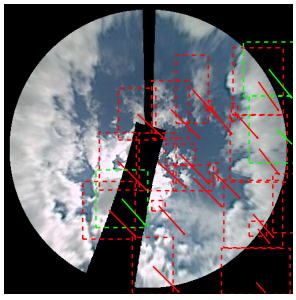


(f) Aggregation at $TSI3_t$

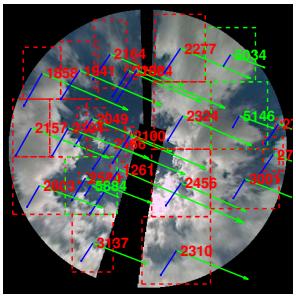


(g) Two-layer stitched view

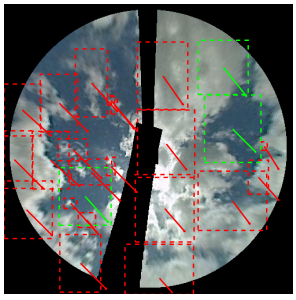
Figure 4-8: Determination of two cloud layers corresponding to the images shown in Figure 4-7. (a), (b), and (c) show the matching results for $(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t)$. The estimated heights are labeled in (b). (d), (e), and (f) are marked with blocks indicating the two wind fields (red and green). The arrows in the images from TSI1 and TSI3 represent motions, whereas the arrows in the images corresponding to TSI2 represent the $TSI_2 \rightarrow TSI_1$ (green) and $TSI_2 \rightarrow TSI_3$ (blue) displacement vectors.



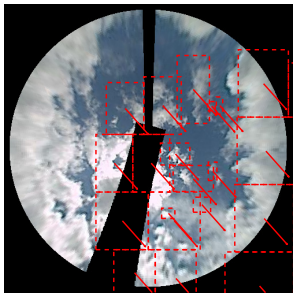
(a) Max. φ at $TSI1_t$



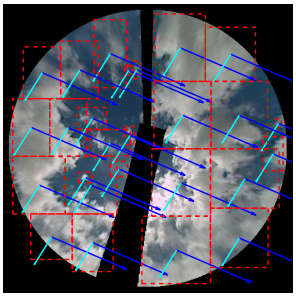
(b) Max. φ at $TSI2_t$



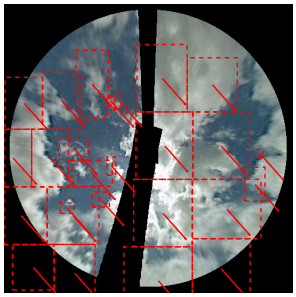
(c) Max. φ at $TSI3_t$



(d) Single WF at $TSI1_t$



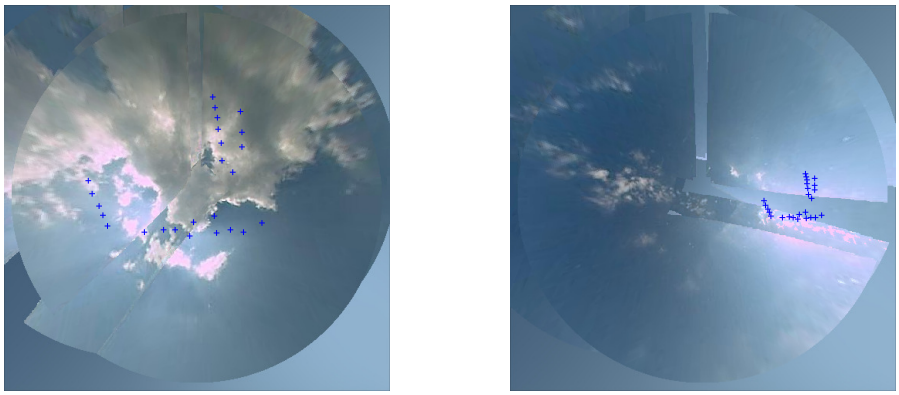
(e) Single WF at $TSI2_t$



(f) Single WF at $TSI3_t$

(g) Single-layer stitched view based on $TSI2_t$

Figure 4-9: The single layer detected from the nine images shown in Figure 4-5. (a), (b), and (c) show the tracking results for $(\mathbf{d}_m^{1,2,t}, \mathbf{v}_m^t)$. The regions marked with green boxes in (b) display obvious bias due to the boundary effect. (d), (e), and (f) show the single-layer field, (h^t, v_x^t, v_x^t) = (2130 m, 8 px, 9 px) marked in red.



(a) h=1000 at 2:00PM (b) h=3000 at 9:00AM

Figure 4-10: Examples of the locations of the sun-blocking pixels (blue crosses) for the 25 pyranometers in the stitched multi-TSI view.

Table 4.1: Descriptions of four subsets with various cloud and weather conditions. $image_{\#}$: number of TSI images, $k_{\#}$: number of GHI measurements, cf: estimated range of cloud fractions in the sky images, cf_{σ}: mean and standard deviation of the cloud fraction, k: range of the clear-sky index. k_{σ}: the mean and standard deviation of k, exp: observed (ab)normal condition of the TSI images, CBH_{est}: cloud height range, v_{est}: cloud motion in image, $WF_{\#}$: number of cloud layers, $zenith$: solar zenith range, $condition$: cloud conditions.

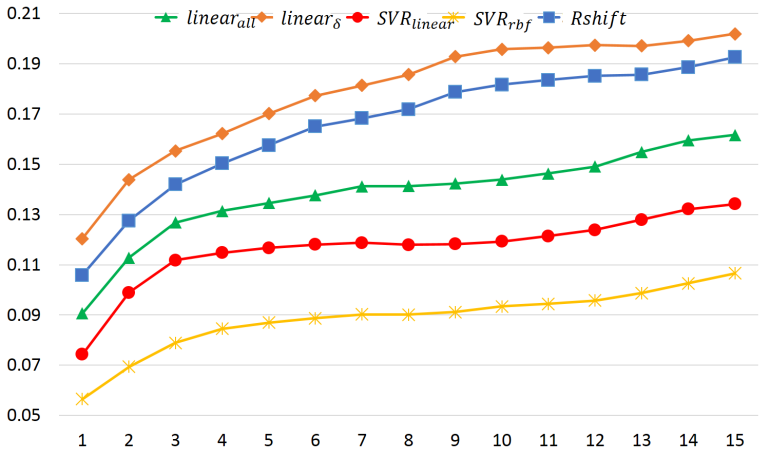
Subset	D_s	D_m	D_o	D_{mix}
$image_{\#}$	2517x3	2520x3	2406x3	2520x3
$k_{\#}$	2517x25	2520x25	2406x25	2520x25
cf	[0.07,0.94]	[0.03,0.94]	[0.94,0.95]	[0,0.95]
cf_{σ}	0.71 ± 0.25	0.70 ± 0.27	0.94 ± 0.01	0.50 ± 0.35
k	[0.11,1]	[0.06,1]	[0.17,1]	[0.11,1]
k_{σ}	$0.46. \pm 0.22$	0.57 ± 0.27	0.63 ± 0.21	0.81 ± 0.23
exp	normal	underexposed	normal	green-dominant
$WF_{\#}$	1	2	2	2
$CBH_{est}(m)$	[1590,2960]	[1890, 4420]	[6020,15730]	[440 12330]
$v_{est}(px/min)$	[36,60]	[6,36]	[0,54]	[0,60]
zenith	$[40°, 57°]$	$[42°, 58°]$	$[41°, 56°]$	$[41°, 57°]$
condition	low,scattered	multi-layer	overcast	mixture

Table 4.2: MAE and RMSE metrics for 1-minute and 5-minute irradiance forecasts. The subscript of each score indicates the subset type used to train and test the model. The subscript "avg" indicates the average performance on the entire dataset.

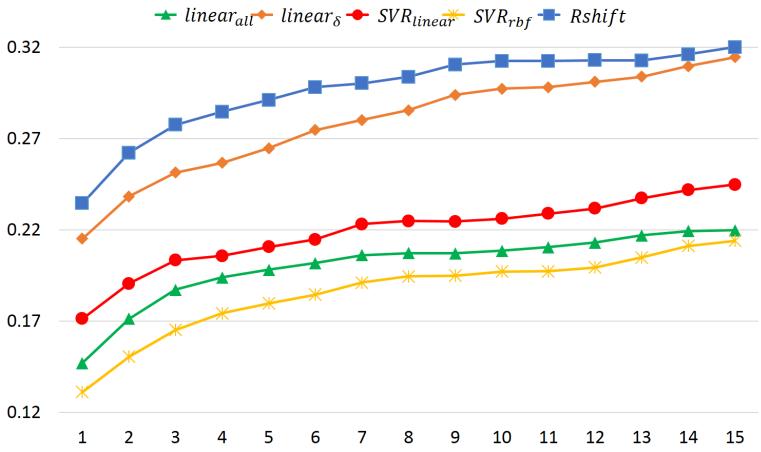
Dataset	R_{shift}		$linear_{\delta}$		$linear_{all}$		SVR_{linear}		SVR_{RBF}	
	1 m	5 m	1 m	5 m	1 m	5 m	1 m	5 m	1 m	5 m
MAE_s	0.16	0.20	0.14	0.20	0.11	0.16	0.10	0.16	0.09	0.14
MAE_m	0.16	0.19	0.14	0.19	0.11	0.15	0.10	0.14	0.08	0.13
MAE_o	0.04	0.08	0.04	0.08	0.03	0.07	0.03	0.07	0.03	0.06
MAE_{mix}	0.11	0.16	0.12	0.17	0.09	0.13	0.07	0.12	0.06	0.09
$\mathbf{MAE_{avg}}$	0.12	0.17	0.12	0.17	0.10	0.16	0.09	0.15	0.07	0.12
$RMSE_s$	0.23	0.27	0.20	0.26	0.15	0.20	0.15	0.20	0.14	0.18
$RMSE_m$	0.26	0.29	0.21	0.28	0.15	0.20	0.15	0.21	0.14	0.20
$RMSE_o$	0.06	0.11	0.06	0.11	0.05	0.09	0.05	0.09	0.04	0.08
$RMSE_{mix}$	0.23	0.29	0.22	0.26	0.15	0.20	0.17	0.21	0.13	0.18
$\mathbf{RMSE_{avg}}$	0.23	0.27	0.21	0.26	0.15	0.21	0.17	0.22	0.13	0.19

Table 4.3: MAE and RMSE metrics for 10-minute and 15-minute irradiance forecasts. '-' indicates that no forecast output is available due to cloud tracking failure. '*' indicates an incomplete dataset that has a low STI value or an average performance influenced by incomplete/empty subsets.

Dataset	R_{shift}		$linear_{\delta}$		$linear_{all}$		SVR_{linear}		SVR_{RBF}	
	10 m	15 m	10 m	15 m	10 m	15 m	10 m	15 m	10 m	15 m
MAE_s	0.27*	-	0.25*	-	0.24*	-	0.16*	-	0.13*	-
MAE_m	0.20	0.21*	0.20	0.21*	0.16	0.18*	0.15	0.17*	0.13	0.15*
MAE_o	0.09	0.11	0.09	0.11	0.08	0.10	0.08	0.09	0.07	0.08
MAE_{mix}	0.18	0.19	0.20	0.20	0.14	0.16	0.12	0.13	0.09	0.11
$\mathbf{MAE_{avg}}$	0.17*	0.17*	0.18*	0.17*	0.17*	0.16*	0.15*	0.15*	0.12*	0.11*
$RMSE_s$	0.33*	-	0.30*	-	0.36*	-	0.21*	-	0.18*	-
$RMSE_m$	0.30	0.30*	0.29	0.30*	0.21	0.23*	0.22	0.24*	0.21	0.23*
$RMSE_o$	0.12	0.15	0.12	0.15	0.11	0.12	0.11	0.13	0.10	0.11
$RMSE_{mix}$	0.31	0.32	0.30	0.31	0.21	0.22	0.23	0.24	0.20	0.21
$\mathbf{RMSE_{avg}}$	0.29*	0.29*	0.28*	0.28*	0.22*	0.22*	0.23*	0.23*	0.21*	0.20*



(a) Average MAE scores for five folds



(b) Average RMSE scores for five folds

Figure 4-11: MAE and RMSE scores for irradiance predictions on the data subset D_{mix} over a time range from one to fifteen minutes.

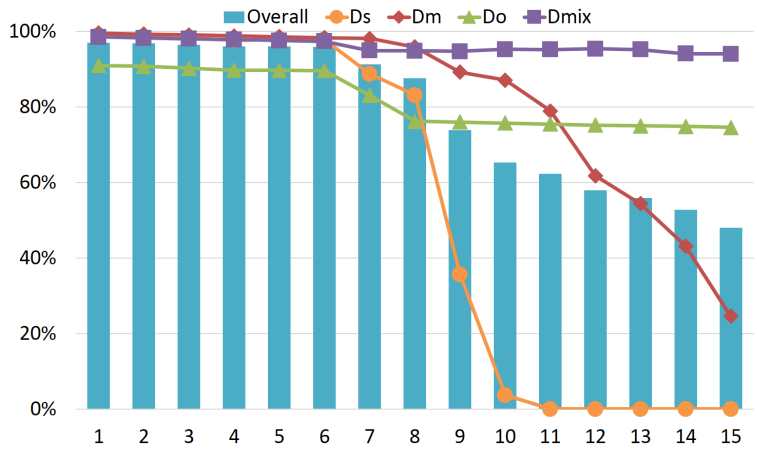


Figure 4-12: Successful Tracking Index (STI) values for the datasets in the 1-15 minute forecasting range. *Overall* represents the results for the entire dataset, which contains all four independent subsets.

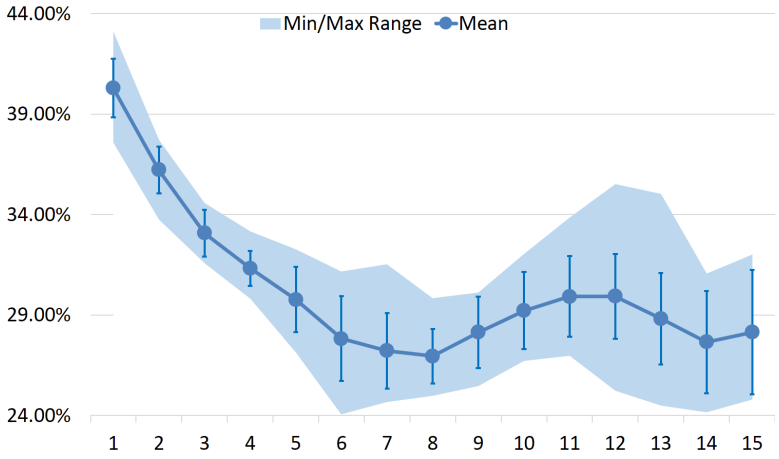
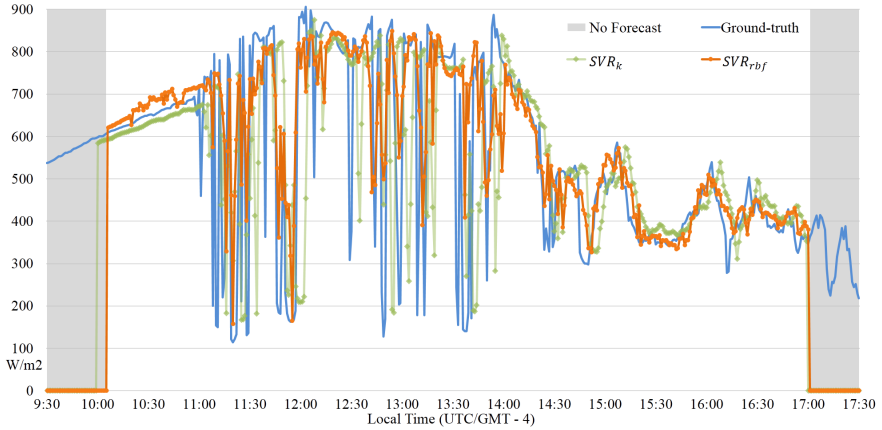
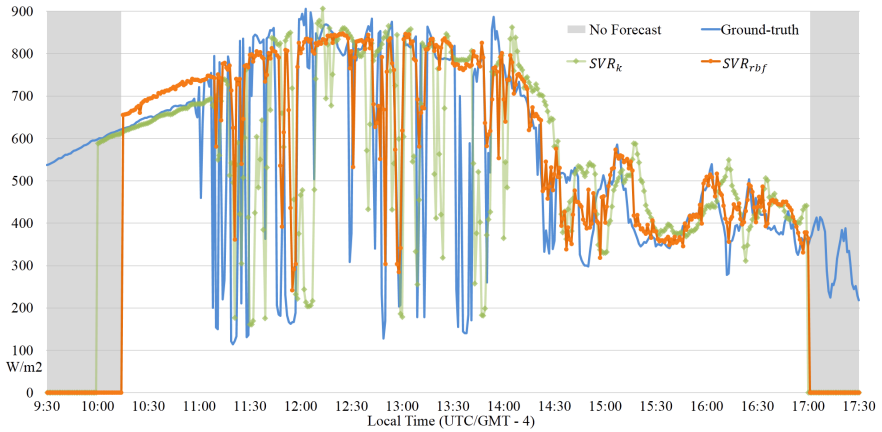


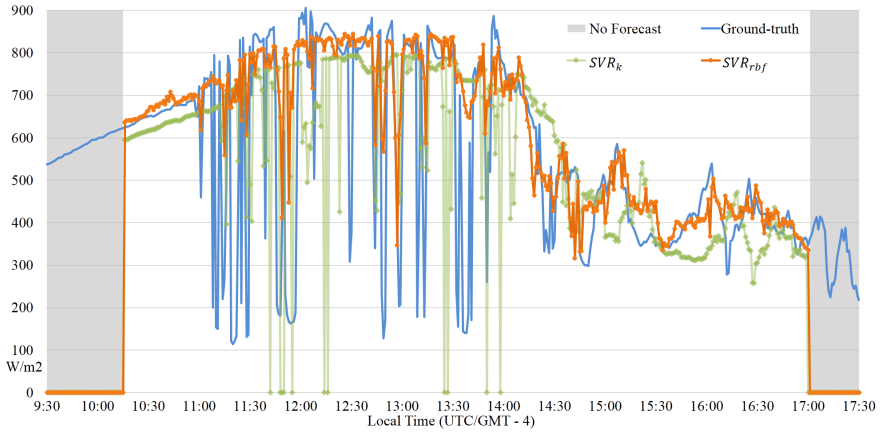
Figure 4-13: Improvements in the MAE ratio achieved by the non-linear SVR_{rbf} model in comparison with the persistent model on all available data. The Min/Max bounds represent the range of the percentage improvement values for all 25 stations. The average performance is denoted by the plotted line, which includes standard deviation bars on either side.



(a) 5-minute forecasts with $RMSE_{rbf} = 145.3$ and $RMSE_k = 213.5$.



(b) 10-minute forecasts with $RMSE_{rbf} = 171.3$ and $RMSE_k = 223.6$.



(c) 15-minute forecasts with $RMSE_{rbf} = 177.5$ and $RMSE_k = 241.0$.

Figure 4-14: Real forecasts based on our new prediction system using SVR_{rbf} and SVR_k. Gray/dark areas with a flat "0" or no forecast value represent data points that are out-of-FOV or correspond to a low zenith angle. $RMSE_{rbf}$ and $RMSE_k$ are the root-mean-square errors of SVR_{rbf} and SVR_k compared with the real GHI values.

Chapter 5

Satellite Image Integration

To predict solar irradiance variability in mesoscale grids and distributed PV system is beyond the capacity of the aforementioned ground-based imaging solutions. In the chapter, we focus on mid-term forecast and introduce our satellite-based framework that predicts solar irradiance between 30 minutes to five hours. Our available image datasets from Geostationary Operational Environmental Satellite (GOES) have erroneous data, missing values or delayed images. To better address these problems and improve prediction accuracy, we focus our effort in the following tasks: 1) Constructing a reliable pipeline to preprocess satellite imagery along with ground-based measurements, such as pyranometer data, for training and validation, 2) Estimating cloud motion using the optical flow framework (OF), and 3) Extracting prominent features from the multi-channel satellite dataset and selecting robust regression model to make predictions. In this chapter, we also include systematic evaluations and comparison between our proposed mid-term model and a baseline model and confirm that the satellite-based solution indeed attains significant improvements over the baseline model for cloud motion estimation and ultimately irradiance prediction.

5.1 Introduction

Cloud tracking and solar forecast on satellite image is not an easy task. Existing models usually focus on visible channel of satellite imager system as it has higher resolution and a good representation for clouds w.r.t visible spectrum. However, relying on visible channel may incur erroneous results in cloud detection and irradiance prediction. The first issue comes from the uncertainties of the cloud. Due to variant cloud shape and unstable motion, cloud tracking and modeling is of higher error rate especially in long time case. An irradiance model must be robust enough to noise and take instability into account during model construction. Second, it is difficult to identify cloud and/or obtain the thickness of cloud only from the visible channels of satellite due to the large spatial resolution (1km x 1km per pixel) and limited observable cloud types. In other words, cloud could be smaller than the grid size or the visible channel shows only a limited spectrum of clouds, resulting in that not all cloud types can be shown on in the visible channels.

Consequently, we try to solve the challenges by adopting new technologies to mitigate the influence of image noise and integrate multiple features to improvement the robustness of cloud tracking and as well as solar prediction models. In detail, we propose to use a framework of optical flow for better motion estimation and adopt Support Vector Regression (SVR) training and modeling the exacting clouds features and incorporate diverse evidence (multiple channel, previous timestamp radiation) to cope with limited information of visible satellite channel. More specifically, our contributions are:

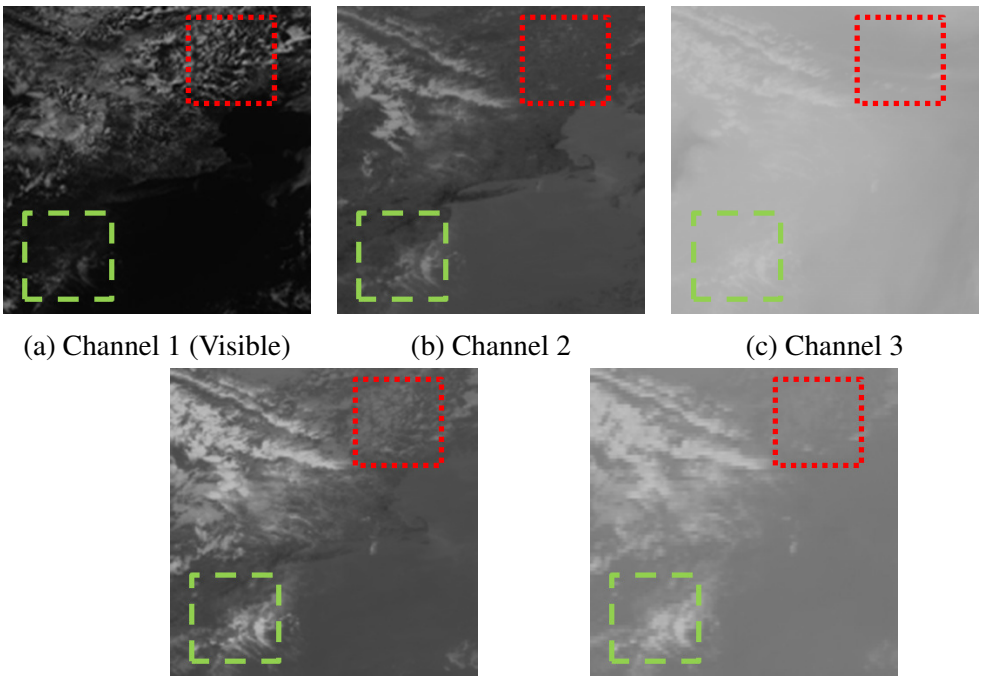
1. **Evaluation of motion estimation methods for satellite** Most of previous works used the block-wise cross correlation [66], [65]. But in the case of satellite images, block-wise motion is unable to track small changes of both cloud shape and motion field. Therefore we propose to use the Optical Flow (OF) motion estimation algorithm which is based on the

gradient of grey-scale change. Compared with traditional algorithms such as HBM, it is more sensitive to small area change and robust even with shape distortion.

2. **Robust regression analysis for irradiance prediction** Most previous works used a variant of the linear regression model [64]. However, this approach is quite sensitive to noise or outliers. We propose Support Vector Regression (SVR) which has several advantages over the ordinary linear regression models. First, SVR ignores outlier points, thereby reducing influence of noise from cloud motion tracking. Second, it can easily model linear and non-linear relations by choosing different kernels, and have better modeling capabilities with a limited number of features. We also provide a simple approach to properly normalize solar irradiance level to remove diurnal and seasonal affects, so that the subsequent regression analysis could be simpler.

3. **Combining multiple evidences** To overcome the limitations of the visible channel, we incorporate other four available channels from satellite imager system to sense the radiant in the different spectrum range. Second, we added the radiation level of the previous timestamp as an additional feature to SVR model have better context information.

4. **Systematic evaluation of whole pipeline** We did systematic cross validations on 6 months data, and evaluated 1) the effects of different cloud motion estimation algorithms, 2) the different irradiation modeling performance from the ground truth cloud image to irradiation level, 3) the effects of the integrated system from the motion estimation to irradiation prediction of up to five hours, and 4) the effects of diverse evidences related to irradiation prediction.



(a) Channel 1 (Visible) (b) Channel 2 (c) Channel 3

(d) Channel 4 (e) Channel 6

Figure 5-1: Multi-channel view: Visible channel fails to capture clouds in the left-bottom box but other channels do not show clouds on right-top box

5.2 Satellite Multi-channel Imagery

5.2.1 Geostationary Satellite Imagery

Satellite images are widely adopted for cloud detection, classification and motion estimation [109] [145]. However most of prior works focused on visible channel. Though some of them proposed to use other spectral channels to identify clouds [107] and model the irradiance at ground-level [106], the utilization of multiple channels of images for solar forecast is not addressed. With the advance of satellite imaging system, the satellite is able to obtain generated image from radiometer every 15 minutes with high resolution.

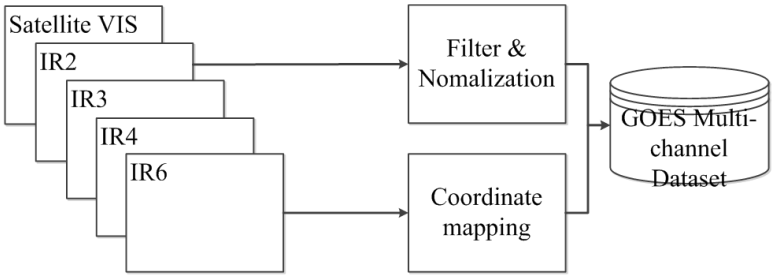
In our work, we collected the images from geostationary satellite for cloud observation, motion tracking, and the following solar forecast. Though other satellites such as POES series are able to provide higher resolution, geostation-

ary satellite is better for all-time stable cloud observation and accurate cloud movement estimation. In detail, we collected 15min-per-frame images from GOES project [146]. As shown in the Figure 5-1, five spectral channels are synchronously collected for the eastern coast of U.S. It is seen that from visible channel is able to present detailed texture of clouds in red (right-up) box however fails to capture the clouds within green (left-bottom) box. With different spectrum sensitivity, other four channels on the contrary are able to observe clouds in the green box. Therefore, we integrate all five channels together in our satellite-based solar forecast system to extract cloud features and build irradiance models.

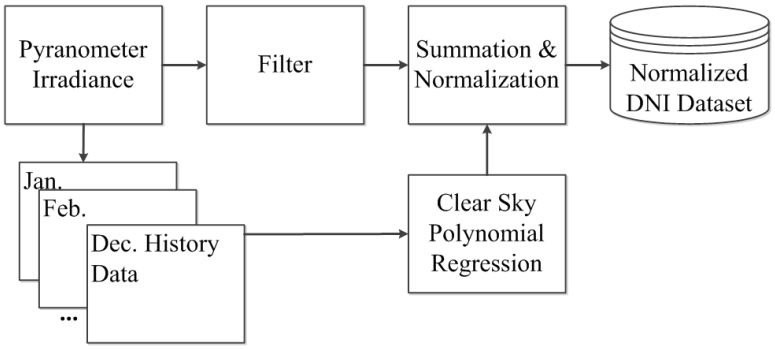
5.2.2 Data Preprocessing

Due to the limitation of remote sensing techniques, we have to use all available evidences to reduce abnormalities brought from satellite data. Therefore we expand preprocessing work to multi-channel instead of just visible channel of satellite. From the dataset collected by GOES project [146], we found several frequent error patterns: 1) black rasters of multi-spectral images due to failure of sensing or raw data processing, 2) luminance variation, especially on visible channel, and 3) missing channel data. We used empirical filters such as mean filter to fill out black raster lines and bad-frame filter to remove low quality or missed frames. We normalized the brightness according to solar zenith angle. The overview of data preprocessing is shown in Figure 5-2a.

Another key step in preprocessing pipeline in the Figure 5-2b is handling radiation data measured by Pyranometer. There are two concerns: 1) radiation level normalization from bell shape curve to uniform level to avoid daily and seasonal effects and 2) different temporal resolution resolving between satellite (30 minutes) and Pyranometer (one second). In previous works, clear sky irradiance is calculated by clear sky models which are based on atmospheric



(a) Satellite Preprocessing



(b) Irradiance Preprocessing

Figure 5-2: Data Preprocessing Framework

parameters such as O2, CO2, Ozone, water vapour and aerosol optical thickness [64]. However, such models are sensitive to the location and ask additional input variables. To address this concern, we propose to use polynomial regression to generate the monthly clear sky radiation curve shown in Figure 5-3. Then we calculate normalized radiation at t using the average of $[t-N, t+N]$ minutes radiation divided by the average of clear sky values. Normalization using average over $2N$ minutes minimizes the influence of temporal resolution mismatch and smoothes out short-term local irradiance fluctuation.

5.3 Motion Estimation

The sequence of satellite images provide motion information of cloud field and allow us to predict distribution in near future. For this, the motion estimation algorithms play a crucial role for irradiance forecasting. In image processing

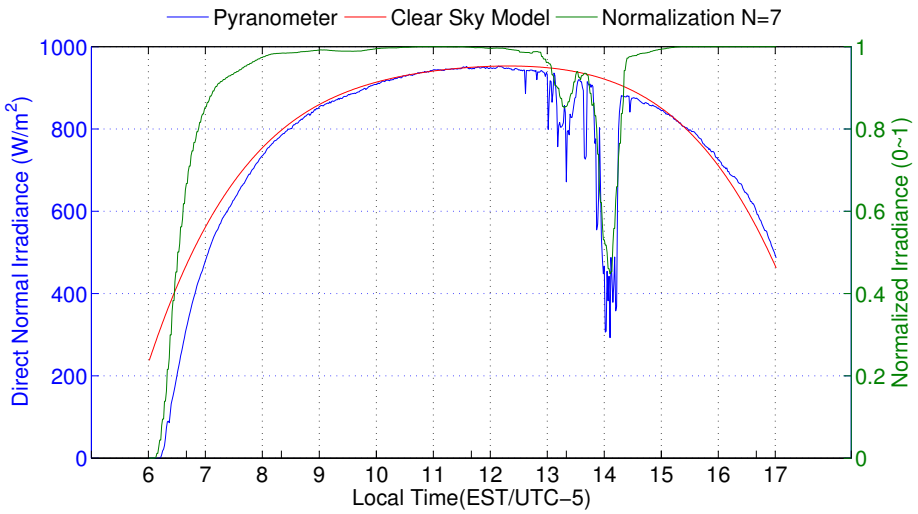


Figure 5-3: Irradiance normalization and clear sky profile on March 1st, 2012

and multimedia field, a well studied approach is Hierarchical Block Matching(HBM) which is on the basis of similarity check among blocks. It is widely used in video coding and compressing as its properties of fast speed and high compressing ratio. As to accuracy, the performance lies on region size and feature matching which assumes consistence of image segmentation. For satellite image case, cloud variation should be considered in smaller scale instead of blocks. Therefore we propose to use pixel-wise approach.

Optical flow (OF) motion estimation is a branch of methodologies that utilize the gradient of image. Under the assumption of constant illuminance, displacement of pixels will be estimated following gradient change. The implementation of OF idea has a lot of variations as gradient of image is defined differently. In our approach, we choose Lucas-Kanade Optical Flow(LKOF) [147] as the tracking method. For robustness, we implement a pipeline which builds the pyramid of image. The framework of pipeline is presented in Figure 5-5.

In general, visible channel image is firstly scaled to different resolution levels to build pyramid of image (i.e. 1000x1000 and 500x500). For each level, motion vectors are computed in turn, following gradient starting from pre-knowledge of motion. The start motion vectors are the output of motion

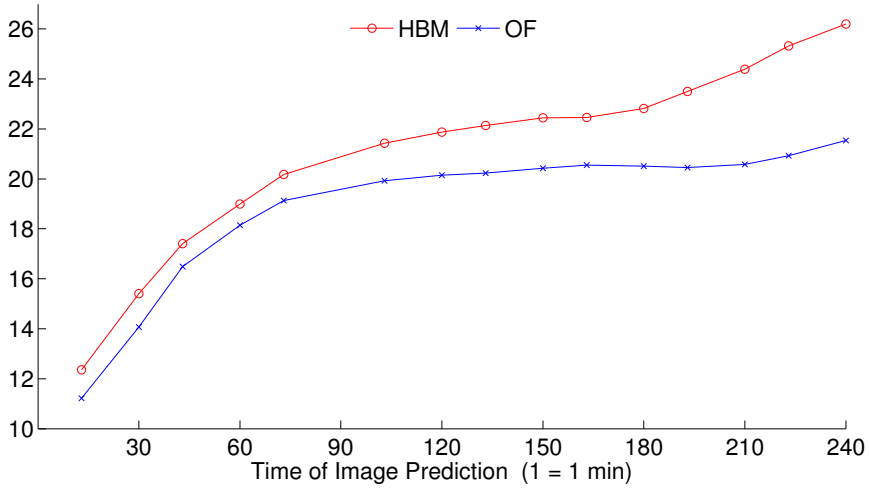


Figure 5-4: Motion estimation result comparison against ground truth images (MAE) : OF is much better than HBM with longer prediction time

from last level in pyramid. When the last level completes motion vector extraction, we utilize this as pre-knowledge of motion on original size. Then the final motion vector is found by seeking for local minima following the gradient recursively. As image prediction based on the pixel-wise motion vector will generate "black hole" due to moving of pixels. we apply mean filter on the OF pipeline output to be our OF_{mean} method so as to reduce information loss.

Intuitively, OF pipeline and HBM outputs can be compared using predicted images. As is shown in Figure 5-6, in one hour image sequences, OF method is much more robust in terms of tiny texture changes and clouds diminishing. HBM is suffering over-estimated problems as in dash-block case and more information loss such in dot-block case. In terms of whole image deviation from ground truth, the HBM Mean Absolute Error(MAE) in grayscale is much more than Optical Flow approach, especially when prediction time is over 3 hours (Figure 5-4). In addition to pixel-based or image quality based evaluation, we evaluate irradiance prediction performance using two approaches compared to ground measured solar radiation.

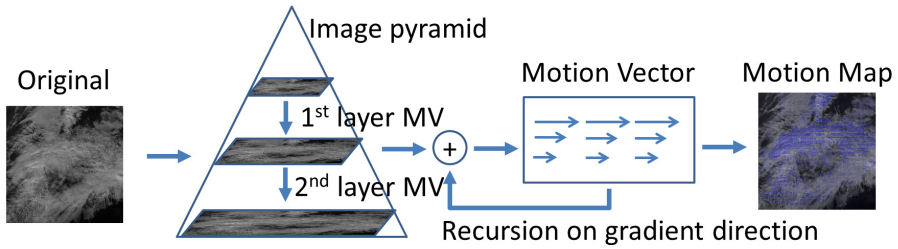


Figure 5-5: Optical flow motion estimation pipeline

5.4 Mid-term Prediction Models

Given the satellite images, we may model irradiance from the cloud properties at a point of our interest on the ground. Our goal is to find a method that 1) utilizes multiple evidences such as multi-channel features and previous timestamp irradiance and 2) is robust to the noise coming up from preprocessing and motion estimation. In this Section, we will investigate the simplest Cloud Index (CI), Linear Regression (LR, ALR) and Support Vector Regression (SVR).

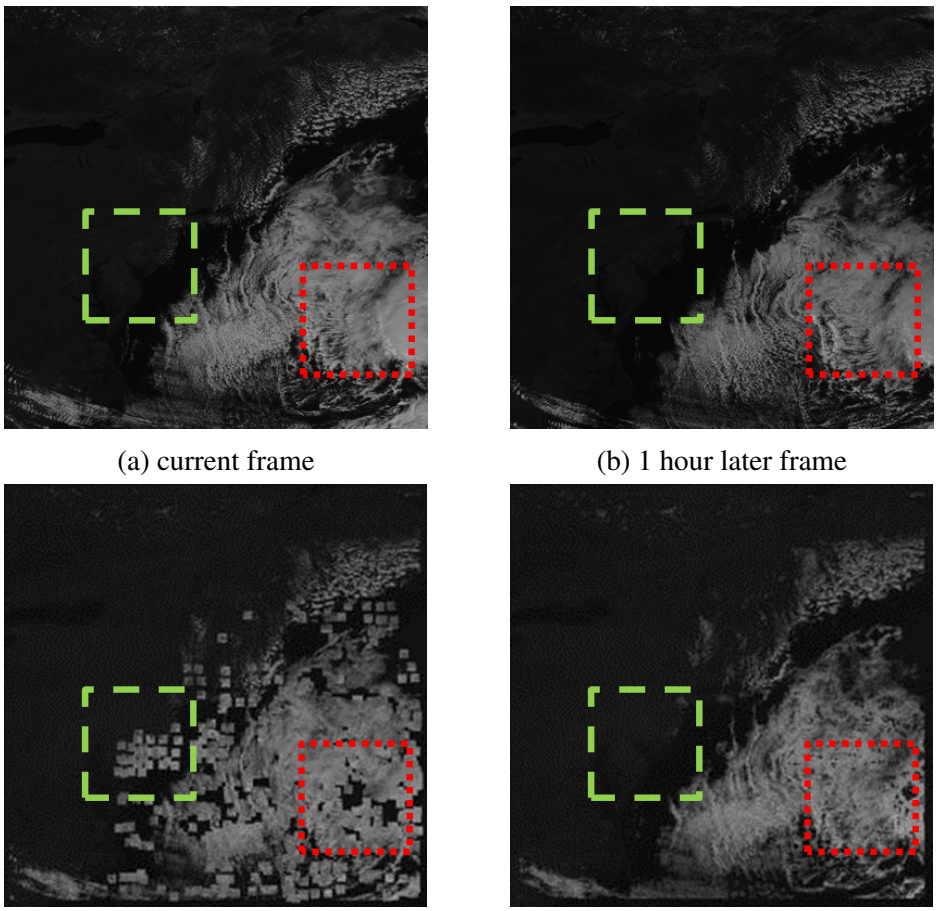
Cloud Index (CI) Cloud Index (CI) [64] is a widely used estimation model on the basis of linear relation between single channel (visible channel) with radiation. CI for a sample time point i is defined as:

$$CI_i = \frac{x_i - bound_{min}}{bound_{max} - bound_{min}} \tag{5.1}$$

where $x_i \in [0, 255]$ is a visible channel value and $bound_{min}$ and $bound_{max}$ are minimum and maximum observed x value. In other words, CI_i is a normalized cloud level between zero and one. Given CI_i, the irradiance is calculated:

$$y_i = w \cdot (1 - CI_i) + b \tag{5.2}$$

where w is the maximum radiation level which is one in our case due to irradiance normalization and b is called the compensation value which is learned by the training data. Cloud Index model is just the observed visual channel output



(a) current frame (b) 1 hour later frame

(c) HBM (d) OF

Figure 5-6: Image Estimation of 1 hour later from (a). With reference to ground-truth frame (b), (d) captures more texture changes in red window and is more robust to noise in green window than (c).

to be rescaled to the radiation level with the compensation value to correct bias. There are couple of problems of this simple method. First of all, CI model will suffer from illuminance changes with solar zenith angle. Due to this problem, we preprocessed the satellite image to be normalized with zenith angle for all of our model input. Second, we can only fit the bias term, so that it can not be fitted well with the data. Third, CI model used only single channel, which means we can not combine the other available evidences.

Linear Regression (LR, ALR) The straight forward extension of CI model is Linear Regression (LR) and Aggregate Linear Regression (ALR) using mul-

tiple channels as extra evidence.

$$y_i = \mathbf{w} \cdot \mathbf{x_i} + b \tag{5.3}$$

where $\mathbf{w} \in R^d$ is a row weight vector and $\mathbf{x_i} \in R^d$ is a column vector for the input, and b is the intercept. The advantages over CI model are that LR models may include multiple evidence as linear relationship and both $\mathbf{w}$ and b are learned from the training data to be better fitted. But LR is susceptible to the noise because of the least square objective function to find optimal $\mathbf{w}$.

$$\min_{\mathbf{w},b} \sum_i (\mathbf{w} \cdot \mathbf{x_i} + b - y_i)^2 \tag{5.4}$$

If there are outlier points, then $\mathbf{w}$ will be overfitted.

Support Vector Regression (SVR) To avoid such overfitting from noisy data, we propose to use Support Vector Regression (SVR) as our solution.

The basic idea of SVR is the regression error to be within ε. If it can not be learned within ε bound, the slack variables allow us to fit the model out of ε bound but the slack variable has to be minimized with C regularization term, Equation 4.32. Since it bounds errors to be within ε, it is much more robust and it is easy to extend non-linear relationship using kernel trick, which projects data into high dimensional space, so that we could model non-linear relationship with linear model [144]. We used RBF (Radial Basis Function) kernel:

$$k(\mathbf{w}', x) = \phi(\mathbf{w}') \cdot \phi(\mathbf{x}) = e^{-\frac{\|\mathbf{w}' - \mathbf{x}\|^2}{2\sigma^2}} \tag{5.5}$$

where σ is a tuning parameter for the smoothness of RBF kernel.

For ALR and SVR, we considered not only visible channel but also the remaining four channels and the previous timestamp irradiance value as the context information. By combining multiple channels, we can overcome the limitations of visible channel and have better cloud distribution information from the

other channels. We will evaluate the different evident effects in the next Section.

5.5 Experiment Results

5.5.1 Dataset

We collected satellite dataset from April 1st 2012 to November 1st 2012, covering partial spring, full summer, and most autumn. The radiation data is from Pyranometer on-site measurement from Brookhaven National Laboratory. For both satellite images and Pyranometer measurements, the raw input data has been preprocessed by removing data points which have 1) any bad-frame of multi-channels 2) failure of ground radiation sensor 3) multi-channel timestamp mismatch and 4) low solar angles. In total, we extract daytime dataset which has 8477 frames for each channel and radiation timestamp continuously. Each radiation data point is a result of normalization and average value in range of 15 minutes. The detailed information of dataset is summarized in Table 5.1.

Table 5.1: Satellite dataset

Dimension	size	temporal resolution	spatial resolution
Channel 1	8477	15min,30min	1km x 1km
Channel 2,3,4,6	8477	15min,30min	4km x 4km
radiation	8477	15min	NA

5.5.2 Parameter Tuning and Evaluation Metrics

For whole dataset, we used five-fold cross validation to split training and testing and used five-fold cross validation again to tune model parameters within training data. In our experiment, Mean Square Error (MSE) and Mean Absolute Error (MAE) are used to evaluate the predicted model accuracy. They are defined as:

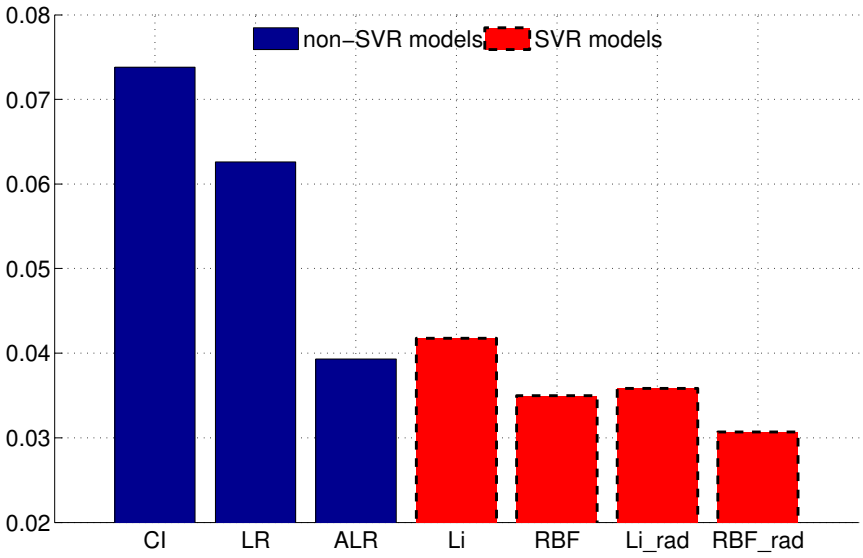


Figure 5-7: MSE Plot of 5-fold cross validation. SVR models show less amounts of errors than non-SVR models in general.

$$MAE = \frac{1}{N}\sum_i |\hat{y}_i - y_i| \tag{5.6}$$

$$MSE = \frac{1}{N}\sum_i (\hat{y}_i - y_i)^2 \tag{5.7}$$

where $\hat{y}_i$ is the predicted radiation level, and $y_i \in [0,1]$ is the ground truth radiation level. Note that the radiation level is normalized by clear sky irradiance value.

5.5.3 Model Comparison

We evaluate the irradiance modeling power using ground truth images. As we described, we compare *CI*, *LR* with single channel, *ALR* which is *LR* with multi-channel, SVR_{Li} - SVR with multi-channel and linear kernel, SVR_{RBF} - SVR with multi-channel and RBF kernel, SVR_{Li_rad} - SVR_{Li} with the previous timestamp radiation level, SVR_{RBF_rad} - SVR_{RBF} with the previous timestamp radiation level.

First of all, Figure 5-7 and 5-8 shows that CI is worse than *LR*, which con-

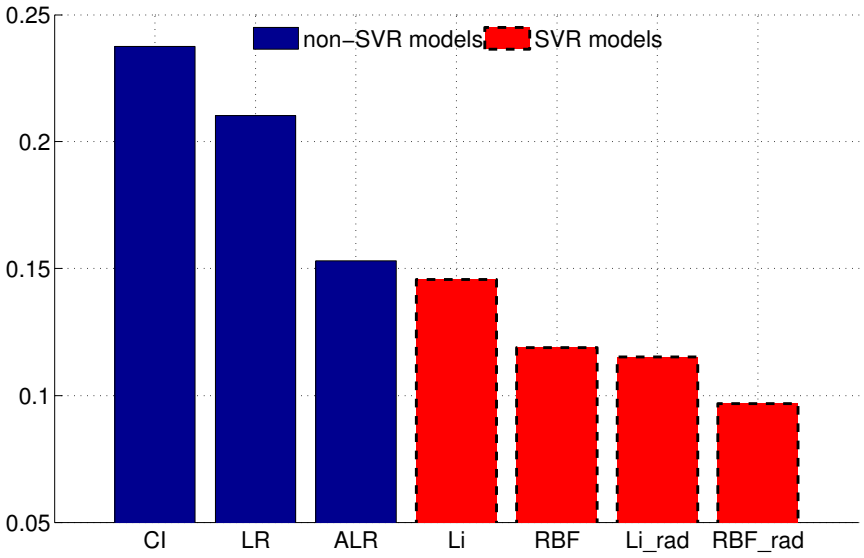


Figure 5-8: MAE Plot of 5-fold cross validation. SVR models show less amounts of errors especially combined with previous radiation feature.

firms that the weight coefficient learning help to learn better model. Second, multi-channel data significantly decrease error rates (*LR* vs *ALR*) and the previous timestamp radiation level is helpful to decrease error rates as well. *SVR* models are better than non-SVR models and non-linear kernel SVR models are even better. Since SVR_{RBF_rad} showed the best performance, we will use it for motion estimation analysis.

5.5.4 Motion Estimation Evaluation

To evaluate motion estimation in the context of radiation forecast applications, the motion estimation output of *HBM* and OF_{mean} are compared as the input to SVR models. We evaluated them from 0.5 to 5 hours of SVR_{RBF_rad} forecast results. As a reference, we also add no motion estimation as our baseline (0 hour image as motion estimated image) and the original *OF* which has no mean filter. The MAE score is shown in Figure 5-9. No motion estimation shows obviously the worst prediction quality but *HBM* shows slightly improvements over the baseline. Both *OF* and OF_{mean} shows significantly better prediction

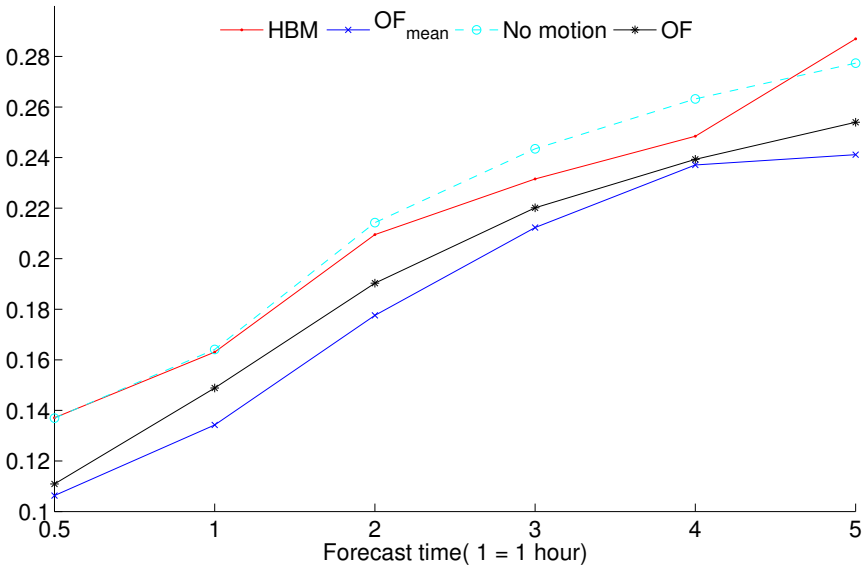


Figure 5-9: Motion estimation analysis using SVR_{RBF_rad} model (MAE). Both OF_{mean} and OF show less amounts of errors than HBM in 5 hours prediction

quality and the mean filter is helpful across different forecast time. It confirms that OF with mean filter is the best choice for satellite motion estimation and we will use OF_{mean} as our motion estimation method for next analysis.

5.5.5 Irradiance Prediction Evaluation

We analyze whether our model estimation results (without any forecast) are still hold or not. In other words, we investigate the noise effects of the motion estimation on the irradiance prediction problem.

Since there are more errors due to motion estimation, *LR* or *ALR* shows even worse results than *CI* (Figure 5-10) after four or five hours, which is different from the model estimation results and expected as we discussed in previous section. However, the multi-channel of *ALR* is still helpful to get much less error rates than *LR*. In case of SVR, SVR_{Li} is the worst among SVR models but with the additional radiation feature, it shows the best performance on or after three hours, which is also different from the model estimation results. RBF kernel with the radiation feature is still the best results under two hours but as

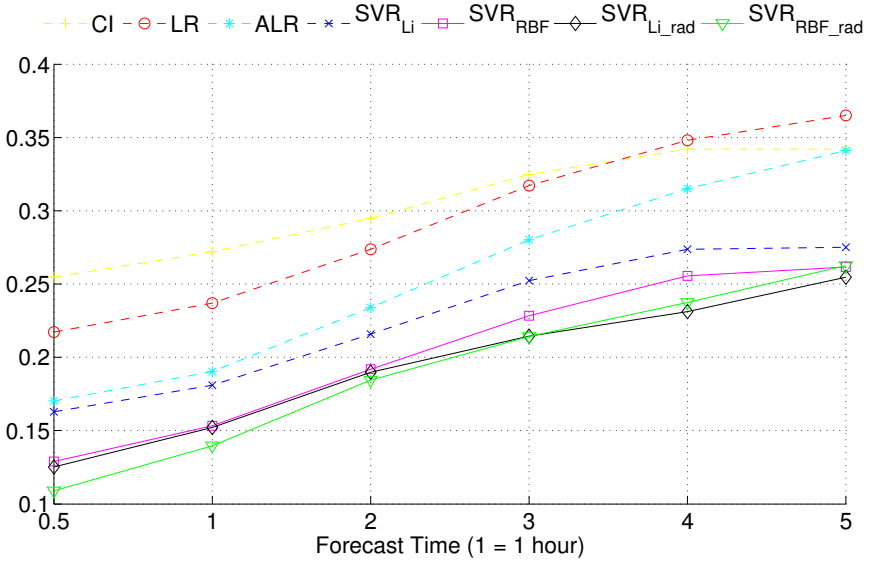


Figure 5-10: MAE score of 7 forecasting models using OF_{mean}.

we have more and more errors from the motion estimation, it is not as robust as SVR_{Li_rad}. If we have an easy day for motion estimation, SVR_{RBF_rad} can be a good choice even after three hours but if it is difficult to do good motion estimation and/or longer time forecasting, then SVR_{Li_rad} will be our choice.

5.5.6 Experiments Findings and Discussion

We introduced two types of error metrics (MAE and MSE) in real data experiments and quantitative model evaluation. Our evaluation results demonstrated that by integrating multiple satellite channels, such as infrared and moisture, our system improves the accuracy of cloud tracking and consistently enhances the overall performance of all regression-based prediction models (ALR, SVR_{Li}, SVR_{RBF}, SVR_{Li_rad}, and SVR_{RBF_rad}). However, the cost of integrating multiple channels is not negligible. We must choose a proper preprocessing technique for satellite data and irradiance measurements to ensure the optimal performance. More importantly, a cost-effective solution for motion tracking and estimation is required to handle such a large scale continental cloud system as obtained from GOES. The block-matching (BM) method produced promising results in

ground-based sky imagers, but it is not appropriate for satellites. Compared with the BM method, the optical flow model indeed lowered the MAE error by at least 20% for mid-term solar forecast. Therefore, we chose the optical flow method because in general it was more efficient and effective in processing multi-channel satellite data than other alternatives.

Furthermore, we also noticed some persistent features, such as historical irradiance values, can improve prediction performance. We undertook some experiments to compare the models with or without persistent features, and uncovered that the mixture models with persistent features always perform much better than those without. Especially for a short forecasting horizon, the mixture models accomplished impressive reduction in error rates. However, once we increased the forecasting horizon, we observed that the "weights" of persistent features faded and the mixture models tended to perform similarly to those without the contribution of persistent features. This observation coincides with the fact that the persistent model is effective for a short time range and deteriorates rapidly with an increasing forecasting range.

5.6 Summary

In this chapter, a new mid-term forecast system is developed with innovations on both modeling and predicting aspect. To construct a robust model, we investigate five channels of geostationary satellite, estimated robust cloud motion at pixel level, and extract cloud information for SVR modeling and forecasting.

In cloud motion estimation, we evaluated both block-wise and optical methods and generate the best optical model that outperforms others in both image analysis and solar forecast evaluation. To choose and compare the best satellite models, we systematically evaluate CI, LR, and SVR methods in various forecast horizons, covering both linear and non-linear approaches. In hours' forecasting, we find that SVR related models significantly improve the predic-

tion accuracy than non-SVR approaches. Although noise from cloud prediction increase exponentially with time, the SVR with non-linear kernel and the previous radiation level still show the best results up to two hours but the SVR with linear kernel and the previous radiation level could be a good candidate after three or more hour prediction ranges. The accuracy improvement is more than 50% in 30 minutes prediction and 10% in 5 hours prediction than the baseline satellite model.

Chapter 6

Conclusion and Future work

This chapter summarizes our accomplishments and explores the future research directions in image processing and solar forecast. By utilizing new imaging techniques, our work demonstrate great potentials in designing and implementing image-based solar forecast. To achieve the optimal performance, we make contributes to several critical components in the end-to-end prediction pipeline, i.e., image-based cloud tracking, feature extraction, and machine learning based regression. Furthermore, we propose to integrate multiple image datasets into solar forecast system to cover a wide range of forecast including short-term and mid-term.

6.1 Conclusion

The advances in imaging technologies and the growing interests in solar energy give rise to a pressing need of designing a reliable image-based solar forecast that extracts cloud spatio-temporal information from images and thereby regresses from high-quality cloud features to irradiance values. To cater this need and build an end-to-end solution for reliable predictions with a wide time range between minutes and hours ahead, we make a series of endeavors in image preprocessing, motion tracking, multi-camera integration, and multi-layer

determination based on various types of input images. Our salient contributions include:

1. We developed a reliable cloud detection pipeline that is based on a supervised classier and applies the histogram equalization technique to correct abnormal images due to camera artifacts or wrong configuration. Our research significantly improved the accuracy of cloud mask extraction under various cloud, weather, and lighting conditions.

2. On the basis of prior works on cloud tracking in sky imagery, we designed a hybrid cloud motion tracking model to combine block matching and optical flow. This new model is adaptive to complex cloud activities and has its advantages in determining local deformations of clouds, extracting cloud layers with dominant motion patterns, and denoising the resulting motion field with customized motion filters.

3. We proposed a comprehensive framework that simulates multiple types of cloud activities (i.e., translative movements, deformation, shrink and expansion), and generates synthetic image sequences containing various cloud activities and multiple levels of Gaussian noise. The simulation framework was used to evaluate multiple motion tracking models under different scenarios, e.g., cloud deformations and corrupted cloud images.

4. We devised a short-term solar forecast system utilizing multiple ground-based sky cameras. The system includes several innovations, i.e., emulating human eyes to detect the depth of far-field objects, intelligently incorporating cloud observations from multiple view angles, applying spatio-temporal correlations to track clouds across multiple cameras and timestamps, and utilizing a stereoscopy calculation to obtain the 3-D spatial coordinates of a cloud. In addition, our system detect multi-layer motions via clustering, which is only possible because of 3-D cloud infor-

mation. Furthermore, we investigated robust feature extraction and irradiance models. Compared with a single-camera model, our proposed system has a significantly larger field of view, enables 3D cloud tracking, and obtains better forecast results. We implemented an end-to-end pipeline that integrates these new capabilities and provides real production forecast for Long Island Solar Farm (LISF).

5. We implemented a mid-term solar irradiance forecast using the multi-spectral images of geostationary satellite. Different from prior works that only make use of visible channel, we adopted five channels of satellite to extract cloud information of different spectrums. To minimize input noise, we designed a preprocessing step to normalize irradiance measurements and select high-quality satellite images. As a result, our system entails several robust irradiance models and thereby significantly improves the performance of hours-ahead prediction.

6.2 Future Work

6.2.1 Image Processing

The performance of solar forecast hinges on the quality of sky and satellite images. One future direction is in the field of image processing, i.e., raw image noise removal, accurate cloud mask detection, textural features extractions. In this thesis work, we already utilized supervised classifier to identify cloud at a pixel level. This approach is sensitive to image noise. Semi-supervised and unsupervised learning methods have great potential in identifying cloud pixels in sky and satellite images. Even object detection for sky images is still challenging,several object detection algorithms, including sparse coding and optical flow recognition, are promising in tracking cloud movement and reconstructing three-dimensional cloud images.

6.2.2 Robust Motion Estimation

The majority of short-term and mid-term forecasts assume that the movement of cloud pixel and object is time-invariant and previously detected motion vectors are simply used for predicting the future location of clouds. Consequently, any deviation from real cloud motion leads to falsely predicted sun-occlusions and errors in solar power forecasts. Furthermore, this type of error accumulates with the forecast time horizon. To better represent real cloud movements, we must treat cloud motion as a function of time that follows the trend of prior motions and is governed by cloud physics, instead of a persistent value from historical data. Existing image-based motion vectors are prone to error and can not be directly translated to cloud movements in real space. A promising research direction is to integrate more ground-based observations into motion estimation, e.g., wind profiler, LIDAR, to eliminate outliers and reconstruct three-dimensional motions.

6.2.3 Feature Mining

The image features extracted from cloud images are critical to the performance of all types of solar prediction models. In our current studies, we exploited 23 prominent features from sky images and six features from the different channels of satellite. In future, we plan to extract textural and spectral features, e.g., output of Garbor filter on images, to describe the characteristics of sun-occlusion clouds. Previous works [148] already demonstrated the efficacy of certain statistical features. We will derive similar statistical properties during cloud detection and tracking, and evaluate their potentials in tracking clouds that are out-of-FOV and extending the forecast horizon that currently is constrained by the physical limitation of tracking device.

6.2.4 Solar Forecast Model

Another potential research direction is to improve the models that output irradiance values based on input feature values. In addition to the regression-based models used in this thesis, we will investigate neural network models, such as artificial neural network (ANN) [149] and deep neural network (DNN) [150] [151] [152] for directly predicting the output of solar power plants.

6.2.5 Integration of Heterogeneous Data Sources

To mitigate the limitation of localized view of TSI and low resolution of a satellite system, we will explore available meteorology data sources. Furthermore, one potential approach is to create an ensemble of multiple state-of-art models of different forecast categories. For instance, weather prediction model (NWP) helps categorize cloud distribution in a long time range. We can integrate the NWP results into our two models and use the long term information of cloud distribution to refine our cloud motion detection and forecast. We will also take advantage of other measurements from ground devices, for example, ground-level thermometer to measure the ambient temperature of PV panels that serves as a good index and efficiency indicator for current gross solar energy reception, LIDAR and RADAR that provide detailed measurements of cloud vertical distribution and atmospheric parameters.

Bibliography

[1] Renewables 2015 Global Status Report. http://www.ren21.net/wp-content/uploads/2015/07/GSR2015_KeyFindings_lowres.pdf. Accessed: 2016-03-01.

[2] Daryl M Chapin, CS Fuller, and GL Pearson. A new silicon p-n junction photocell for converting solar radiation into electrical power. *Journal of Applied Physics*, 25(5):676–677, 1954.

[3] Solar Market Insight 2015 Q4. http://www.seia.org/research-resources/solar-market-insight-2015-q4. Accessed: 2016-03-01.

[4] CJ Hanley et al. Solar energy grid integration systems–energy storage. *SAND2008-4247, Sandia National Laboratories, Albuquerque, NM*, 2008.

[5] Samuel R West, Daniel Rowe, Saad Sayeef, and Adam Berry. Short-term irradiance forecasting using skycams: Motivation and development. *Solar Energy*, 110:188–207, 2014.

[6] DG Steyn, TR Oke, JE Hay, and JL Knox. On scales in meteorology and climatology. *Clim Bull*, 39:1–8, 1981.

[7] Sue Ellen Haupt and Branko Kosovic. Big data and machine learning for applied weather forecasts: Forecasting solar power for utility operations. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 496–501. IEEE, 2015.

[8] Jovan Bebic. *Power system planning: emerging practices suitable for evaluating the impact of high-penetration photovoltaics*. Citeseer, 2008.

[9] JL Bosch and J Kleissl. Cloud motion vectors from a network of ground sensors in a solar power plant. *Solar Energy*, 95:13–20, 2013.

[10] Zhenzhou Peng, Shinjae Yoo, Dantong Yu, and Dong Huang. Solar irradiance forecast system based on geostationary satellite. In *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, pages 708–713. IEEE, 2013.

[11] Zhenzhou Peng, Shinjae Yoo, Dantong Yu, Dong Huang, Paul Kalb, and John Heiser. 3d cloud detection and tracking for solar forecast using multiple sky imagers. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 512–517. ACM, 2014.

[12] Zhenzhou Peng, Dantong Yu, Dong Huang, John Heiser, Shinjae Yoo, and Paul Kalb. 3d cloud detection and tracking system for solar forecast using multiple sky imagers. *Solar Energy*, 118(0):496 – 519, 2015.

[13] Qingyong Li, Weitao Lu, and Jun Yang. A hybrid thresholding algorithm for cloud detection on ground-based color images. *Journal of atmospheric and oceanic technology*, 28(10):1286–1296, 2011.

[14] RW Johnson, TL Koehler, and JE Shields. A multistation set of whole sky imagers and a preliminary assessment of the emerging data base. In *Proc. Cloud Impacts on DOD Operations and Systems1988 Workshop*, pages 159–162, 1988.

[15] Yinghao Chu, Hugo TC Pedro, Lukas Nonnenmacher, Rich H Inman, Zhouyi Liao, and Carlos FM Coimbra. A smart image-based cloud detection system for intrahour solar irradiance forecasts. *Journal of Atmospheric and Oceanic Technology*, 31(9):1995–2007, 2014.

[16] Yair Weiss. Smoothness in layers: Motion segmentation using nonparametric mixture estimation. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 520–526. IEEE, 1997.

[17] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[18] Thomas M Hamill and Thomas Nehrkorn. A short-term cloud forecast scheme using cross correlations. *Weather and Forecasting*, 8(4):401–411, 1993.

[19] Patrick Héas, Etienne Mémin, Nicolas Papadakis, and André Szantai. Layered estimation of atmospheric mesoscale dynamics from satellite imagery. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(12):4087–4104, 2007.

[20] Patrick Héas and Etienne Mémin. Three-dimensional motion estimation of atmospheric layers from image sequences. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(8):2385–2396, 2008.

[21] Hao Huang, Shinjae Yoo, Dantong Yu, Dong Huang, and Hong Qin. Cloud motion detection for short term solar power prediction. In *Proceedings of ICML 2011 Workshop on Machine Learning for Global Challenges.*, 2011.

[22] J. Alonso-Montesinos, F.J. Batlles, and C. Portillo. Solar irradiance forecasting at one-minute intervals for different sky conditions using sky camera images. *Energy Conversion and Management*, 105:1166 – 1177, 2015.

[23] JE Shields, ME Karr, TP Tooman, DH Sowle, and ST Moore. The whole sky imager–a year of progress. In *Eighth Atmospheric Radiation Measurement (ARM) Science Team Meeting, Tucson, Arizona*, pages 23–27, 1998.

[24] JE Gil. Experimental set up to study the cloud radiative effects on uvb at granada (spain). In *Geophysical Research Abstracts*, volume 7, page 00320, 2005.

[25] A Cazorla, FJ Olmo, and L Alados-Arboledas. Development of a sky imager for cloud cover assessment. *JOSA A*, 25(1):29–39, 2008.

[26] CN Long, DW Slater, and Tim P Tooman. *Total sky imager model 880 status and testing results*. Pacific Northwest National Laboratory, 2001.

[27] VR Morris. Total sky imager (tsi) handbook. Technical report, ARM TR-017, 2005.

[28] S Quesada-Ruiz, Y Chu, J Tovar-Pescador, HTC Pedro, and CFM Coimbra. Cloud-tracking methodology for intra-hour dni forecasting. *Solar Energy*, 102:267–275, 2014.

[29] B Urquhart, B Kurtz, E Dahlin, M Ghonima, JE Shields, and J Kleissl. Development of a sky imaging system for short-term solar power forecasting. *Atmospheric Measurement Techniques Discussions*, 7(5):4859–4907, 2014.

[30] Handa Yang, Ben Kurtz, Dung Nguyen, Bryan Urquhart, Chi Wai Chow, Mohamed Ghonima, and Jan Kleissl. Solar irradiance forecasting using a ground-based sky imager developed at uc san diego. *Solar Energy*, 103:502–524, 2014.

[31] Saad Sayeef and Sam West. Very short-term solar forecasting using inexpensive fisheye camera sky-imagery. In *The 52nd Annual Conference of the Australian Solar Council, Melbourne, Australia*, 2014.

[32] Shanhui Sun, Jan Ernst, Archana Sapkota, Eberhard Ritzhaupt-Kleissl, Jeremy Wiles, Joachim Bamberger, and Terrence Chen. Short term cloud coverage prediction using ground based all sky imager. In *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, pages 121–126. IEEE, 2014.

[33] J. Nou, R. Chauvin, S. Thil, J. Eynard, and S. Grieu. Clear-sky irradiance model for real-time sky imager application. *Energy Procedia*, 69:1999 – 2008, 2015. International Conference on Concentrating Solar Power and Chemical Energy Systems, SolarPACES 2014.

[34] G Pfister, RL McKenzie, JB Liley, A Thomas, BW Forgan, and Charles N Long. Cloud coverage based on all-sky imaging and its impact on surface solar irradiance. *Journal of Applied Meteorology*, 42(10):1421–1434, 2003.

[35] JE Shields, RW Johnson, and TL Koehler. Automated whole sky imaging systems for cloud field assessment. In *Fourth Symposium on Global Change Studies, American Meteorological Society*, pages 228–231, 1993.

[36] MS Ghonima, B Urquhart, CW Chow, JE Shields, A Cazorla, and J Kleissl. A method for cloud detection and opacity classification based on ground based sky imagery. *Atmospheric Measurement Techniques*, 5(11):2881–2892, 2012.

[37] Charles N Long, Jeff M Sabburg, Josep Calbó, and David Pagès. Retrieving cloud characteristics from ground-based daytime color all-sky images. *Journal of Atmospheric and Oceanic Technology*, 23(5):633–652, 2006.

[38] MP Souza-Echer, EB Pereira, LS Bins, and MAR Andrade. A simple method for the assessment of the cloud cover state in high-latitude regions by a ground-based digital camera. *Journal of Atmospheric and Oceanic Technology*, 23(3):437–447, 2006.

[39] KA Buch, Chen-Hui Sun, and LR Thorne. Cloud classification using whole-sky imager data. In *of the 5th Atmospheric Radiation Measurement Science Team Meeting, San Diego, CA, USA*, pages 19–23, 1995.

[40] Maneesha Singh and Matt Glennen. Automated ground-based cloud recognition. *Pattern analysis and applications*, 8(3):258–271, 2005.

[41] Josep Calbo and Jeff Sabburg. Feature extraction from whole-sky ground-based images for cloud-type recognition. *Journal of Atmospheric and Oceanic Technology*, 25(1):3–14, 2008.

[42] Anna Heinle, Andreas Macke, and Anand Srivastav. Automatic cloud classification of whole sky images. *Atmospheric Measurement Techniques Discussions*, 3(1):269–299, 2010.

[43] A Kazantzidis, P Tzoumanikas, AF Bais, S Fotopoulos, and G Economou. Cloud detection and classification with the use of whole-sky ground-based images. *Atmospheric Research*, 113:80–88, 2012.

[44] Renxiang Li, Bing Zeng, and Ming L Liou. A new three-step search algorithm for block motion estimation. *Circuits and Systems for Video Technology, IEEE Transactions on*, 4(4):438–442, 1994.

[45] Mohammed Sayed. A fast architecture for exhaustive search block matching algorithm with mpeg-4 applications. In *Electronics, Circuits, and Systems, 2009. ICECS 2009. 16th IEEE International Conference on*, pages 787–790. IEEE, 2009.

[46] Chi Wai Chow, Bryan Urquhart, Matthew Lave, Anthony Dominguez, Jan Kleissl, Janet Shields, and Byron Washom. Intra-hour forecasting with a total sky imager at the uc san diego solar energy testbed. *Solar Energy*, 85(11):2881–2893, 2011.

[47] Hao Huang, Shinjae Yoo, Dantong Yu, Dong Huang, and Hong Qin. Correlation and local feature based cloud motion estimation. In *Proceedings of the Twelfth International Workshop on Multimedia Data Mining*, pages 1–9. ACM, 2012.

[48] Hao Huang, Jin Xu, Zhenzhou Peng, Shinjae Yoo, Dantong Yu, Dong Huang, and Hong Qin. Cloud motion estimation for short term solar irradiation prediction. In *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, pages 696–701. IEEE, 2013.

[49] Philip Wood-Bradley, José Zapata, and John Pye. Cloud tracking with optical flow for short-term solar forecasting. In *50th conference of the australian solar energy society, Melbourne*, 2012.

[50] David Bernecker, Christian Riess, Elli Angelopoulou, and Joachim Hornegger. Towards improving solar irradiance forecasts with methods from computer vision. *Pattern Recognition*, 2012.

[51] Spyretta Golemati, John S Stoitsis, Aimilia Gastounioti, Alexandros C Dimopoulos, Vassiliki Koropouli, and Konstantina S Nikita. Comparison of block matching and differential methods for motion analysis of the carotid artery wall from ultrasound images. *Information Technology in Biomedicine, IEEE Transactions on*, 16(5):852–858, 2012.

[52] Yinghao Chu, Hugo TC Pedro, and Carlos FM Coimbra. Hybrid intra-hour dni forecasts with sky image processing enhanced by stochastic learning. *Solar Energy*, 98:592–603, 2013.

[53] Ricardo Marquez, Vesselin G Gueorguiev, and Carlos FM Coimbra. Forecasting of global horizontal irradiance using sky cover indices. *Journal of Solar Energy Engineering*, 135(1):011017, 2013.

[54] S Janjai, P Pankaew, and J Laksanaboonsong. A model for calculating hourly global solar radiation from satellite data in the tropics. *Applied Energy*, 86(9):1450–1457, 2009.

[55] Pierre Ineichen and Richard Perez. A new airmass independent formulation for the linke turbidity coefficient. *Solar Energy*, 73(3):151–157, 2002.

[56] Werner Möser and Ehrhard Raschke. Incident solar radiation over europe estimated from meteosat data. *Journal of Climate and Applied Meteorology*, 23(1):166–170, 1984.

[57] CG Justus, MV Paris, and JD Tarpley. Satellite-measured insolation in the united states, mexico, and south america. *Remote Sensing of Environment*, 20(1):57–83, 1986.

[58] HS Jang, KY Bae, HS Park, and DK Sung. Effect of aggregation for multi-site photovoltaic (pv) farms. In *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 628–633. IEEE, 2015.

[59] Richard Perez, Sergey Kivalov, James Schlemmer, Karl Hemker, David Renné, and Thomas E Hoff. Validation of short and medium term operational solar radiation forecasts in the us. *Solar Energy*, 84(12):2161–2172, 2010.

[60] Richard Perez, Kathleen Moore, Steve Wilcox, David Renné, and Antoine Zelenka. Forecasting solar radiation–preliminary evaluation of an approach based upon the national forecast database. *Solar Energy*, 81(6):809–812, 2007.

[61] Annette Hammer, Detlev Heinemann, Carsten Hoyer, Rolf Kuhlemann, Elke Lorenz, Richard Müller, and Hans Georg Beyer. Solar energy assessment using remote sensing technologies. *Remote Sensing of Environment*, 86(3):423–432, 2003.

[62] A Zelenka, R Perez, R Seals, and D Renné. Effective accuracy of satellite-derived hourly irradiances. *Theoretical and applied climatology*, 62(3-4):199–207, 1999.

[63] David S Renne, Richard Perez, Antoine Zelenka, CH Whitlock, and RC DiPasquale. Use of weather and climate research satellites for estimating solar resources. *BIOPROCESS TECHNOLOGY*, 13:171–240, 1999.

[64] Richard Perez, Pierre Ineichen, Kathy Moore, Marek Kmiecik, Cyril Chain, Ray George, and Frank Vignola. A new operational model for satellite-derived irradiances: description and validation. *Solar Energy*, 73(5):307–317, 2002.

[65] John A Leese, Charles S Novak, and V Ray Taylor. The determination of cloud pattern motions from geosynchronous satellite image data. *Pattern Recognition*, 2(4):279–292, 1970.

[66] S Cote and ARL Tatnall. A neural network-based method for tracking features from satellitesensor images. *International Journal of Remote Sensing*, 16(18):3695–3701, 1995.

[67] Adrian N Evans. Cloud motion analysis using multichannel correlation-relaxation labeling. *Geoscience and Remote Sensing Letters, IEEE*, 3(3):392–396, 2006.

[68] Thomas Corpetti, Patrick Heas, E Mémin, and Nicolas Papadakis. Variational pressure image assimilation for atmospheric motion estimation. In *Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International*, volume 2, pages II–505. IEEE, 2008.

[69] Berthold K Horn and Brian G Schunck. Determining optical flow. In *1981 Technical Symposium East*, pages 319–331. International Society for Optics and Photonics, 1981.

[70] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, pages 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

[71] Chi Wai Chow, Serge Belongie, and Jan Kleissl. Cloud motion and stability estimation for intra-hour solar forecasting. *Solar Energy*, 115:645–655, 2015.

[72] Michael J Black and Paul Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer vision and image understanding*, 63(1):75–104, 1996.

[73] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(3):500–513, 2011.

[74] Glenn Rutledge, Eugene Legg, and Paul Menzel. Operational production of winds from cloud motions. *Global and Planetary Change*, 4(1):141–150, 1991.

[75] Harold S Stone, Michael T Orchard, E-C Chang, and Stephen A Martucci. A fast direct fourier-based algorithm for subpixel registration of images. *Geoscience and Remote Sensing, IEEE Transactions on*, 39(10):2235–2243, 2001.

[76] David Bernecker, Christian Riess, Elli Angelopoulou, and Joachim Hornegger. Continuous short-term irradiance forecasts using sky images. *Solar Energy*, 110:303–315, 2014.

[77] J-P Thirion. Image matching as a diffusion process: an analogy with maxwell's demons. *Medical image analysis*, 2(3):243–260, 1998.

[78] Aristeidis Sotiras, Christos Davatzikos, and Nikos Paragios. Deformable medical image registration: A survey. *Medical Imaging, IEEE Transactions on*, 32(7):1153–1190, 2013.

[79] Ricardo Marquez and Carlos FM Coimbra. Intra-hour dni forecasting based on cloud tracking image analysis. *Solar Energy*, 91:327–336, 2013.

[80] Nobuhito Mori and Kuang-An Chang. Introduction to mpiv. *User manual and program available online at http://sauron. civil. eng. osaka-cu. ac. jp/˜ mori/softwares/mpiv*, 2003.

[81] Jian Chen, Jie Tian, Noah Lee, Jian Zheng, RTheodore Smith, and Andrew F Laine. A partial intensity invariant feature descriptor for multimodal retinal image registration. *Biomedical Engineering, IEEE Transactions on*, 57(7):1707–1718, 2010.

[82] Abdou Tankari Mahamadou, Bailo Camara Mamadou, Dakyo Brayima, and Nichita Cristian. Ultracapacitors and batteries integration for power fluctuations mitigation in wind-pv-diesel hybrid system. *International journal of renewable energy research (IJRER)*, 1(2):86–95, 2011.

[83] K Stefferud, J Kleissl, and J Schoene. Solar forecasting and variability analyses using sky camera cloud detection & motion vectors. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–6. IEEE, 2012.

[84] Elke Lorenz, Annette Hammer, and Detlev Heinemann. Short term forecasting of solar radiation based on satellite data. In *EUROSUN2004 (ISES Europe Solar Congress)*, pages 841–848, 2004.

[85] Rich H Inman, Hugo TC Pedro, and Carlos FM Coimbra. Solar forecasting methods for renewable energy integration. *Progress in energy and combustion science*, 39(6):535–576, 2013.

[86] Juan L Bosch, Yuehai Zheng, and Jan Kleissl. Deriving cloud velocity from an array of solar radiation measurements. *Solar Energy*, 87:196–203, 2013.

[87] M Lipperheide, JL Bosch, and J Kleissl. Embedded nowcasting method using cloud speed persistence for a photovoltaic power plant. *Solar Energy*, 112:232–238, 2015.

[88] Jonathan R Stroud, Michael L Stein, Barry M Lesht, David J Schwab, and Dmitry Beletsky. An ensemble kalman filter and smoother for satellite data assimilation. *Journal of the American Statistical Association*, 105(491):978–990, 2010.

[89] Yinghao Chu, Bryan Urquhart, Seyyed MI Gohari, Hugo TC Pedro, Jan Kleissl, and Carlos FM Coimbra. Short-term reforecasting of power output from a 48 mwe solar pv plant. *Solar Energy*, 112:68–77, 2015.

[90] Jin Xu, Shinjae Yoo, Dantong Yu, Dong Huang, John Heiser, and Paul Kalb. Solar irradiance forecasting using multi-layer cloud tracking and numerical weather prediction. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, SAC '15, New York, NY, USA, 2015. ACM.

[91] Chia-Lin Fu and Hsu-Yung Cheng. Predicting solar irradiance with all-sky image features via regression. *Solar Energy*, 97:537–550, 2013.

[92] Andrew A Lacis and James Hansen. A parameterization for the absorption of solar radiation in the earth's atmosphere. *Journal of the Atmospheric Sciences*, 31(1):118–133, 1974.

[93] Rolf Stuhlmann, Martin Rieland, Ehrhard Raschke, et al. An improvement of the igmk model to derive total and diffuse solar radiation at the surface from satellite data. *Journal of Applied Meteorology*, 29(7):586–603, 1990.

[94] Detlev Heinemann, Elke Lorenz, and Marco Girodo. Forecasting of solar radiation. *) Solar energy resource management for electricity generation from local level to global scale. Nova Science Publishers, New York*, 2006.

[95] Annette Hammer, Detlev Heinemann, Elke Lorenz, and Bertram Lückehe. Short-term forecasting of solar radiation: a statistical approach using satellite data. *Solar Energy*, 67(1):139–150, 1999.

[96] Richard Perez, Pierre Ineichen, Marek Kmiecik, Kathleen Moore, David Renne, and Ray George. Producing satellite-derived irradiances in complex arid terrain. *Solar Energy*, 77(4):367–371, 2004.

[97] Ulrich Görsdorf, Axel Seifert, Volker Lehmann, and Martin Köhler. Cloud statistics and nwp-model validation based on long term measurements of a 35 ghz radar. In *35th Conference on Radar Meteorology*, pages 26–30, 2011.

[98] Dazhi Yang, Panida Jirutitijaroen, and Wilfred M Walsh. Hourly solar irradiance time series forecasting using cloud cover index. *Solar Energy*, 2012.

[99] Johannes Schmetz. Towards a surface radiation climatology: Retrieval of downward irradiances from satellites. *Atmospheric Research*, 23(3):287–321, 1989.

[100] Pierre Ineichen and Richard Perez. Derivation of cloud index from geostationary satellites and application to the production of solar irradiance and daylight illuminance data. *Theoretical and Applied Climatology*, 64(1-2):119–130, 1999.

[101] Luis F Zarzalejo, Lourdes Ramirez, and Jesus Polo. Artificial intelligence techniques applied to hourly global irradiance estimation from satellite-derived cloud index. *Energy*, 30(9):1685–1697, 2005.

[102] S Janjai, J Laksanaboonsong, M Nunez, and A Thongsathitya. Development of a method for generating operational solar radiation maps from satellite data for a tropical environment. *Solar Energy*, 78(6):739–751, 2005.

[103] Daniel Cano, Jean-Marie Monget, Michel Albuisson, Hervé Guillard, Nathalie Regas, and Lucien Wald. A method for the determination of the global solar radiation from meteorological satellite data. *Solar Energy*, 37(1):31–39, 1986.

[104] FR Martins, EB Pereira, and SL Abreu. Satellite-derived solar resource maps for brazil under swera project. *Solar Energy*, 81(4):517–528, 2007.

[105] RT Pinker and JA Ewing. Modeling surface solar radiation: Model formulation and validation. *Journal of Applied Meteorology*, 24:389–401, 1985.

[106] Richard Perez, Sergey Kivalov, A Zelenka, J Schlemmer, and K Hemker Jr. Improving the performance of satellite-to-irradiance models using the satellites infrared sensors. In *Proc. of American Solar Energy Societys Annual Conference, Phoenix, AZ*, 2010.

[107] E Ricciardelli, F Romano, and V Cuomo. Physical and statistical approaches for cloud identification using meteosat second generation-spinning enhanced visible and infrared imager data. *Remote Sensing of Environment*, 112(6):2741–2760, 2008.

[108] Sylvain Cros, Olivier Liandrat, Nicolas Sebastien, and Nicolas Schmutz. Extracting cloud motion vectors from satellite images for solar power forecasting. In *Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International*, pages 4123–4126. IEEE, 2014.

[109] H Escrig, FJ Batlles, J Alonso, FM Baena, JL Bosch, IB Salbidegoitia, and JI Burgaleta. Cloud detection, classification and motion estimation using geostationary satellite imagery for cloud cover forecast. *Energy*, 2013.

[110] Luis F Zarzalejo, Jesús Polo, Luis Martín, Lourdes Ramírez, and Bella Espinar. A new statistical approach for deriving global solar radiation from satellite images. *Solar Energy*, 83(4):480–484, 2009.

[111] Ozan Şenkal and Tuncay Kuleli. Estimation of solar radiation over turkey using artificial neural network and satellite data. *Applied Energy*, 86(7):1222–1228, 2009.

[112] Ozan Şenkal. Modeling of solar radiation using remote sensing and artificial neural network in turkey. *Energy*, 35(12):4795–4801, 2010.

[113] Zibo Dong, Dazhi Yang, Thomas Reindl, and Wilfred M Walsh. Satellite image analysis and a hybrid esss/ann model to forecast solar irradiance in the tropics. *Energy Conversion and Management*, 79:66–73, 2014.

[114] Peder Bacher, Henrik Madsen, and Henrik Aalborg Nielsen. Online short-term solar power forecasting. *Solar Energy*, 83(10):1772 – 1783, 2009.

[115] Patrick Mathiesen, Craig Collier, and Jan Kleissl. A high-resolution, cloud-assimilating numerical weather prediction model for solar irradiance forecasting. *Solar Energy*, 92:47–61, 2013.

[116] William C Skamarock, Joseph B Klemp, Jimy Dudhia, David O Gill, Dale M Barker, Wei Wang, and Jordan G Powers. A description of the advanced research wrf version 2. Technical report, DTIC Document, 2005.

[117] Elke Lorenz, Jan Kühnert, and Detlev Heinemann. Overview of irradiance and photovoltaic power prediction. In *Weather Matters for Energy*, pages 429–454. Springer, 2014.

[118] A Cazorla, FJ Olmo, and L Alados-Arboledas. Using a sky imager for aerosol characterization. *Atmospheric Environment*, 42(11):2739–2745, 2008.

[119] Tony Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus-of-attention. *International Journal of Computer Vision*, 11(3):283–318, 1993.

[120] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'95, pages 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[121] Michael Stokes, Mattew Anderson, Srinivasan Chandrasekar, and Ricardo Motta. A standard default color space for the internetsrgb, 1996. *URL http://www. w3. org/Graphics/Color/sRGB*.

[122] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[123] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[124] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE, 2012.

[125] David Fleet and Yair Weiss. Optical flow estimation. In *Handbook of Mathematical Models in Computer Vision*, pages 237–257. Springer, 2006.

[126] Jean-Yves Bouguet. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5:1–10, 2001.

[127] Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014.

[128] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2432–2439. IEEE, 2010.

[129] Robert M Haralock and Linda G Shapiro. *Computer and robot vision*. Addison-Wesley Longman Publishing Co., Inc., 1991.

[130] Ricardo Marquez, Hugo TC Pedro, and Carlos FM Coimbra. Hybrid solar forecasting method uses satellite imaging and ground telemetry as inputs to anns. *Solar Energy*, 92:176–188, 2013.

[131] Eli Shusterman and Meir Feder. Image compression via improved quadtree decomposition algorithms. *Image Processing, IEEE Transactions on*, 3(2):207–215, 1994.

[132] Zhuoyuan Chen, Hailin Jin, Zhe Lin, Sholom Cohen, and Ying Wu. Large displacement optical flow from nearest neighbor fields. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2443–2450. IEEE, 2013.

[133] Kaiming He and Jian Sun. Statistics of patch offsets for image completion. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part II*, ECCV'12, pages 16–29, Berlin, Heidelberg, 2012. Springer-Verlag.

[134] Andreas Wedel, Thomas Pock, Christopher Zach, Horst Bischof, and Daniel Cremers. An improved algorithm for tv-l 1 optical flow. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, pages 23–45. Springer, 2009.

[135] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *Computer Vision-ECCV 2004*, pages 25–36. Springer, 2004.

[136] T. Brox. *From pixels to regions: partial differential equations in image analysis*. PhD thesis, Faculty of Mathematics and Computer Science, Saarland University, Germany, April 2005.

[137] Qi Zhang, Li Xu, and Jiaya Jia. 100+ times faster weighted median filter (wmf). In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2830–2837. IEEE, 2014.

[138] Evgueni Kassianov, Charles N Long, and Jason Christy. Cloud-base-height estimation from paired ground-based hemispherical observations. *Journal of Applied Meteorology*, 44(8):1221–1233, 2005.

[139] Mark C Allmen and W Philip Kegelmeyer Jr. The computation of cloud-base height from paired whole-sky imaging cameras. *Journal of Atmospheric and Oceanic Technology*, 13(1):97–113, 1996.

[140] Dung Andu Nguyen and Jan Kleissl. Stereographic methods for cloud base height determination using two sky imagers. *Solar Energy*, 107:495–509, 2014.

[141] Steven S. Beauchemin and John L. Barron. The computation of optical flow. *ACM Computing Surveys (CSUR)*, 27(3):433–466, 1995.

[142] Stuart Lloyd. Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129–137, 1982.

[143] Harris Drucker, Chris JC Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. Support vector regression machines. *Advances in neural information processing systems*, pages 155–161, 1997.

[144] Alex J Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[145] Johannes Schmetz, Kenneth Holmlund, Joel Hoffman, Bernard Strauss, Brian Mason, Volker Gaertner, Arno Koch, and Leo Van De Berg. Operational cloud-motion winds from meteosat infrared images. *Journal of Applied Meteorology*, 32(7):1206–1225, 1993.

[146] NOAA-NASA. Noaa-nasa goes project science, February 2013.

[147] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence*, 1981.

[148] Jin Xu, Shinjae Yoo, Dantong Yu, Hao Huang, Dong Huang, John Heiser, and Paul Kalb. A stochastic framework for solar irradiance forecasting using condition random field. In *Advances in Knowledge Discovery and Data Mining*, pages 511–524. Springer, 2015.

[149] Tamer Khatib, Azah Mohamed, Kamarulzaman Sopian, and M Mahmoud. Solar energy prediction for malaysia using artificial neural networks. *International Journal of Photoenergy*, 2012, 2012.

[150] Jeffrey Dean. Large-scale deep learning for building intelligent computer systems. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, pages 1–1, New York, NY, USA, 2016. ACM.

[151] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016.

[152] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.