

# Cloud Motion Estimation for Short Term Solar Irradiation Prediction

Hao Huang <sup>#1</sup>, Jin Xu <sup>#2</sup>, Zhenzhou Peng <sup>#3</sup>, Shinjae Yoo <sup>\*4</sup>, Dantong Yu <sup>\*5</sup>, Dong Huang <sup>\*6</sup>, Hong Qin <sup>#7</sup>

<sup>#</sup>*Stony Brook University, 100 Nicolls Road, Stony Brook, NY 11794*

<sup>1,7</sup>{haohuang, qin}@cs.stonybrook.edu, <sup>2</sup>jinxu@ece.sunysb.edu, <sup>3</sup>zhenzhou.peng@stonybrook.edu

<sup>\*</sup>*Brookhaven National Laboratory, 50 Bell Avenue, Upton, NY 11973*

<sup>4</sup>sjyoo@bnl.gov, <sup>5</sup>dtYu@bnl.gov, <sup>6</sup>dhuang@bnl.gov

**Abstract**—Variability of solar energy is the most significant issue for integrating solar energy into the power Grid. There are pressing demands to develop methods to accurately estimate cloud motion that directly affects the stability of solar power output. We propose a solar prediction system that can detect cloud movements from the TSI (total sky imager) images, and then estimate the future cloud position over solar panels and subsequent solar irradiance fluctuations incurred by cloud transients. The experiment studies show that our proposed approach significantly improves the quality of cloud motion estimation within a time window (up to a few minutes) that is sufficient for grid operators to take actions to mitigate the solar power volatility.

## I. INTRODUCTION

Exploiting solar energy is highly challenging because it cannot be used as an on-demand stable power source. Variability in solar source is the biggest hurdle in integrating solar energy into the national Grid. Various solutions, such as backup generators, batteries, power trade and exchange were proposed to mitigate beforehand the undesired random behavior. However these solutions must be built upon the accurate prediction of photovoltaic (PV) output in particular for the time period from a few seconds to minutes that is long enough to wake up the backup substitute sources.

Cloud motion is the primary reason for PV output fluctuation. In essence, short-term PV output prediction can be simplified as cloud motion estimation. A typical cumulus cloud whose lifetime usually lasts only a few minutes is the most important for solar farm scale forecasting. Local weather forecast provides some global cloud coverage information on a daily basis, but is certainly insufficient for our prediction requirement which has small time granularities and spatial resolutions. High resolution cloud-resolving radar can track cloud location, height and motions, but is prohibitively expensive and hence also not practical for solar installations.

Our goal is to provide **cost-effective solar irradiance prediction models** that require no expensive equipment while meeting the accuracy and performance requirements. We address the variability problem of cloud motion with Total Sky Imager (TSI) [8], a cost-effective solution to take hemispheric images of the whole visible sky (Figure 1(a)). We detect cloud motion vectors between TSI images and use them to estimate future cloud motion over the location where solar panels reside. The prediction result of cloud motion can be used to calculate solar radiation level with the predicted location of cloud, Sun position, the timestamps and locations of TSI. A pyranometer (Figure 1(b)) is used to monitor the local level

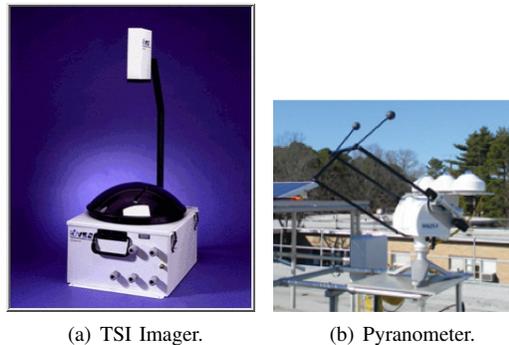


Fig. 1. Two input source equipments.

of solar irradiance on the ground. Its numerical correlation with the cloud motion from the TSI images is captured by our proposed model for predicting the solar radiation values on site. These two devices together offer us a complete yet highly economic input setup for our prediction objectives.

This paper addresses several challenges which are unique to the cloud motion estimation techniques: 1) The proposed prediction algorithm has to be real-time (to wake up the backup devices in time) and in a short term. Longer prediction (hourly and daily) that requires different atmospheric input is not the focus of this paper; 2) Cloud shape changes continuously over time causing instability to cloud motion tracking; 3) Cloud has multiple layers with different velocities and/or directions, which often confuse with the cloud shape changes; 4) TSI loses some information due to the holding arm, shadow band, and boarder of circular images (Figure 2), and it is difficult to track cloud motions around these area; 5) Clouds' direct influence on solar irradiance is reflected in a complex way through the TSI image color patterns and textures.

This paper has the following contributions:

- 1) Our methodology only uses input from TSI and pyranometer without other meteorological devices (eg. humidity, wind velocity). It is spatially and temporally microscopic and structurally different from most of the existing mesoscale approaches [15] [12] [11] [19] [9].
- 2) We use fast cross correlation algorithm for effective and real-time cloud motion tracking (Section IV). It satisfies our real-time speed requirement and is sufficient for the precision of cloud motion tracking in the mean time.
- 3) We propose a few refinement techniques customized for

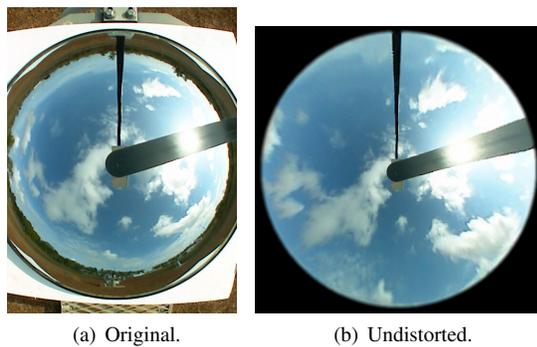


Fig. 2. TSI image undistortion using coordinate transformation.

the cloud motion prediction process (Section V), which can not only work on single layer but also on multi-layer clouds (Section VI).

- 4) Finally, we propose a linear prediction model for short term solar irradiance, and compare the model output with real-life TSI and pyranometer data. Our model has the crucial feature of expendability which is often lacking from other physical and atmospherical motivated models.
- 5) The experiments confirm that our proposed algorithms can significantly improve the accuracy of cloud tracking and solar irradiance estimation over existing methods.

## II. BACKGROUND

Many techniques have been developed to detect object motions. But only a few of them are actually appropriate for cloud motion detection. Arrays of similar partial cloud tend to generate multiple narrow extrema in the correlation surface, which could mislead sequential similarity detection algorithm [1]. Since cloud often does not have a clear boundary, active contour models such as Snakes [2] are difficult to implement. Wavelets and other multi-resolution schemes [7] are not suitable because of the lower resolution of TSI especially near the boundary area, where ideal features are usually hard to identify. Local gradients based methods [5] [10] are robust to deformable objects, but they cannot meet the real-time requirement. Cross correlation is one of the classic optical flow techniques [3] [4] which tries to maximize correlation on block-based image matching. In [15], Marcello et.al. surveyed region matching techniques and concluded that normalized cross correlation is one of the most favored choices for cloud motion detection. The main reason is that cross correlation method makes fewer requirements on the image sequence and has fewer parameters to tune by the users. Its power has also been justified with maximum-likelihood theory [18].

A most recent study with similar goals to our research is [6]. Major drawbacks of this work are: 1) It applies simple correlation-based method and fails to consider multi-layer clouds; 2) It uses only two most-recent frames to predict the future cloud motion; 3) It is less sophisticated without motion vector refinements; 4) Their binary model for cloud condition is too crude for tagging various other types of cloud condition and causes a great barrier to expand their model's applicability and capacity for more complex cloud patterns.

Targeting these problems, our paper introduces novel and practical techniques for both cloud motion and solar irradiance

prediction. Our goal is to deal with the volatile local weather conditions, therefore linear prediction model is used to capture the relations between cloud position/transients and solar irradiance with continuous value. It is desirable especially for building more sophisticated models in the future to utilize more features to meet various prediction requirements.

## III. ALGORITHM FRAMEWORK

We present a schematic overview of our proposed model in Figure 3. The whole process consists of three stages: TSI image preprocessing, cloud motion estimation and solar radiation estimation. The key step of TSI image preprocessing is to map the original non-flat TSI image space into a flat space, so that the motion vectors can be detected with uniform/similar size. We call this step TSI image undistortion (Figure 2). Cloud motion estimation is composed of two motion detection steps and two motion estimation steps. The reason is because of the requirement of filling missing information. Motion estimation quality will be affected by the missing information (causing by holding arm and shadowband of TSI as shown in Figure 2). To overcome this, we use the first round rough motion estimation to fill in missing information on the TSI images. Then the second estimation based on the second detection will produce more accurate sky predictions. Finally, the estimated cloud motion will be used to predict the solar irradiance level in a specific time range. We will build both deterministic and statistical models of cloud variation to characterize cloud transients, and predict the associated solar irradiance.

## IV. FAST AND EFFECTIVE CLOUD MOTION DETECTION

In this Section, we describe the cloud motion detection method which meets real-time requirements and yet is effective enough. In [13], Huang et.al. already investigated two most popular motion vector detection approaches: Phase Correlation (PC) and Cross Correlation (CC), and discussed the advantages and disadvantages for both of them when used in cloud motion detection. Here we describe the fast cross correlation adopted in our research because of its speed and effectiveness.

Given a prior  $N \times N$  sub-frame  $f$ , cross correlation looks for the most correlated current sub-frame  $g$  from search window  $M \times M$  where  $M > N$ . However it is computationally expensive, therefore Lewis [14] presented fast normalized correlation. Since the image can be normalized with zero mean, the numerator of cross correlation formula is a convolution of the image with the reversed feature  $g'(-x, -y)$  where  $g' = g - \bar{g}$  and can be computed using the following equation:

$$FFT^{-1}\{FFT(f)FFT^*(g')\}. \quad (1)$$

Therefore the time complexity for the nominator is  $O(M^2 \cdot \log M)$ . For the denominator, we compute a definite sum from a precomputed running sum [14]. Using two sum tables over the image function  $f$  and image energy  $f^2$ , the sum tables are pre-computed integrals of  $f$  and  $f^2$  over the search image. So for a window of size  $M \times M$ , it can efficiently reduce the arithmetic operations to only three addition/subtraction operations per pixel once the sum tables are established. In other words, the construction of the tables requires approximately  $O(M^2)$  operations, which is far less than  $O(M^2 \cdot N^2)$  required to compute cross correlation at each pixel.

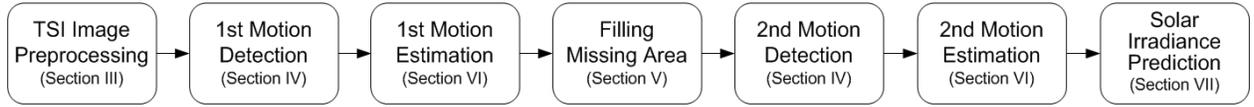


Fig. 3. Overall Cloud Motion and Solar Irradiance Estimation Flow.

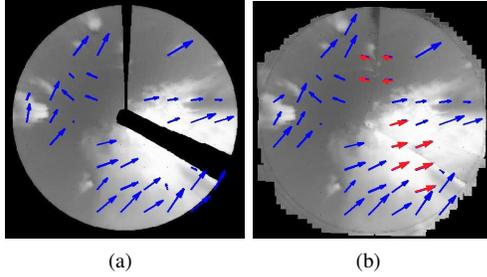


Fig. 4. Filling the missing information (a) with the estimated images. The motion vector detection after filling with the estimated image found additional (red) motion vectors (b) near the holding arm and shadow band.

## V. MOTION VECTOR REFINEMENTS

Although the real-time requirements can be satisfied by our adopted fast cross correlation, it still needs significant improvements for accurate prediction. We summarize the refinements techniques used to improve our motion vector detection.

### A. Filling

The lack of sky information leads to either no motion vector detected, or the detected vectors falsely align with the edges of the empty area. We adopt two different methods to fill in the missing information. The first one is “local mean-filling” used in the first motion detection, in which the empty area will be filled with the local mean of neighbor areas. It removes the majority of the spurious vectors near the holding arm and shadow band. The second method, used in the second motion detection, is to fill with the estimated cloud from the first motion estimation result. Figure 4 shows before and after filling in missing information area and in the latter we could find more accurate motion vectors.

### B. Sequential Cloud Motion Prediction

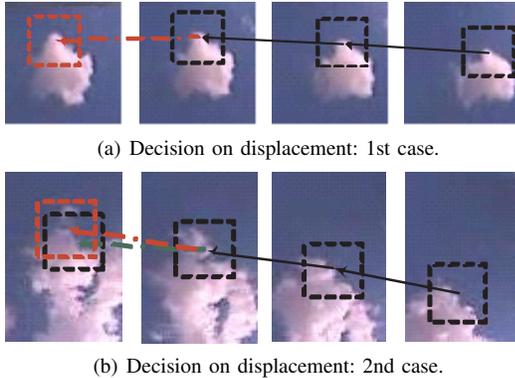


Fig. 5. Cloud Motion Vector Back-tracking.

To cope with the abnormal motion vectors due to cloud shape changes, we try to predict next  $g$  ( $g \geq 1$ ) frame motion

vectors from  $h$  preceding frames. Since it utilizes  $h$  ( $h \geq 2$ ) recent frames, it is more stable compared with using only two recent frames [6]. Our sequential cloud motion prediction model consists of three steps: 1) back-tracing the same block from the current frame to  $h+1$  preceding frames, 2) using a series of vector trend to build a sequential prediction model, and 3) predicting future vectors using the sequential model.

Figure 5 shows the back-tracing step. We back-trace a block starting at  $(i, j)$  of the current image frame  $I^t$  back to the previous frame  $I^{t-1}$  and repeat this process until  $h+1$  previous frames are checked. To increase the stability, we compare the detected back-traced motion vector with the predicted one. If the difference between these two is larger than thresholds, the predicted motion vector is chosen as the final motion vector (Figure 5(a)), otherwise, the detected motion vector is chosen (Figure 5(b)) to allow local motion vector digression resulted from smooth cloud shape changes. Iteratively using the same method on  $h+1$  frames, we can find  $h$  motion vectors namely from  $\vec{v}_{(i,j)}^{t-h}$  to  $\vec{v}_{(i,j)}^t$  for the same block  $(i, j)$ . To denote the motion vector of block  $(i, j)$  in time  $t$ , we use  $\vec{v}^t$  instead of  $\vec{v}_{(i,j)}^t$  for the notational simplicity. The second step is to learn linear prediction models.

$$\hat{\vec{v}}^{t+1} = \sum_{k=1}^h \vec{w}_k \cdot \vec{v}^{t-h+k}, \quad (2)$$

where  $\hat{\vec{v}}^{t+1}$  is the predicted motion vector,  $\vec{v}^{t-h+k}$  are the previous detected vectors, and  $\vec{w}_k$  are the corresponding coefficients. Our goal is to minimize the following loss function:

$$\operatorname{argmin}_W \left\{ \sum_{t=h}^{n-1} \left( \sum_{k=1}^h \vec{w}_k \cdot \vec{v}^{t-h+k} - \vec{v}^{t+1} \right)^2 + \lambda \sum_{k=1}^h \vec{w}_k \cdot \vec{w}_k' \right\}, \quad (3)$$

where  $n$  is the number of training frames and  $\lambda$  is a regularization parameter. To predict next  $g$  vectors, we sequentially predict from  $\hat{\vec{v}}^{t+1}$  to  $\hat{\vec{v}}^{t+g}$  using the following equation.

$$\hat{\vec{v}}^{t+g} = \sum_{k=1}^{h-g-1} \vec{w}_k \cdot \vec{v}^{t-h+k} + \sum_{k=h-g}^{h-1} \vec{w}_k \cdot \hat{\vec{v}}^{t-h+k}. \quad (4)$$

### C. Wind-Field Extraction

Wind-fields are three-dimensional spatial patterns of winds with similar wind speeds. We would like to use the spatial patterns of winds to identify and remove spurious motion vectors. Moreover, since we can only observe the lowest layer of cloud, it is especially important to extract wind-field patterns to adjust the motion detection. Our extraction algorithm relies on three key characteristics of wind-field: frequency of motion vectors, cosine similarity and motion vector length similarity. Algorithm 1 describes the extraction procedures. Algorithm 2 explains how to adjust motion vectors using local and global wind-field. Here local wind-field is extracted within local neighborhood in a single time frame, while global wind-field is extracted from all motion vectors during a given time  $t$ .

---

**Algorithm 1:** Windfield-Extraction( $V, th_{cos}, lth_{ratio}, uth_{ratio}$ )

---

**Input:**  $V$  is a motion vector matrix,  $th_{cos}$  is similarity threshold,  $lth_{ratio}$  and  $uth_{ratio}$  are the lower and upper bound norm length ratio thresholds

**Output:** Wind-field  $WF$

- 1 Let  $\vec{v}_i$  be the  $i^{th}$  largest number of motion vectors in  $V$  ( $\vec{v}_1, \dots, \vec{v}_m$ );
- 2 **foreach**  $\vec{v}_i$  from  $\vec{v}_1$  to  $\vec{v}_m$  **do**
- 3     **if**  $WF = \emptyset$  **then**
- 4         Add  $\vec{v}_i$  to  $WF$  ;
- 5         **continue** to step 2 ;
- 6     **end**
- 7     **foreach**  $\vec{v}_j \in WF$  **do**
- 8          $sim = \cos(\vec{v}_i, \vec{v}_j)$  ;
- 9          $ratio = \|\vec{v}_i\|_2 / \|\vec{v}_j\|_2$  ;
- 10         **if**  $sim \geq th_{cos}$  and  $lth_{ratio} \leq ratio \leq uth_{ratio}$  **then**
- 11             **continue** to step 2 ;
- 12         **end**
- 13     **end**
- 14     **if** the number of  $\vec{v}_i \geq 2\%$  of the number of non-empty motion vectors **then**
- 15         Add  $\vec{v}_i$  to  $WF$  ;
- 16     **end**
- 17 **end**

---

## VI. CLOUD MOTION ESTIMATION

Approaches to estimate single-layer/multi-layer cloud motion are provided in this Section. We make an assumption that the cloud will only change its shape smoothly and maintain a constant velocity in a short enough time period [16]. The core idea of estimating the new cloud image for a future time frame is to use the motion vectors from Section V to project the latest frame ( $t$ ) into a new frame ( $t + g$ ) where  $g \geq 1$ .

As we used image sub-blocks to detect motion vectors, cloud motion estimation is simplified to estimate block-wise movements where each block starting at  $(i, j)$  on time  $t$ , with block size  $m \times m$  ( $I_{(i,j)(i+m-1,j+m-1)}^t$ ), has one motion vector  $\vec{v}_{(i,j)}^t = \langle u, v \rangle$  where  $I^t$  is the image frame at time  $t$ .

$$I_{(i+u,j+v)(i+u+m-1,j+v+m-1)}^{t+1} = I_{(i,j)(i+m-1,j+m-1)}^t \cdot (5)$$

However, Equation 5 will generate mosaic effects between block boundaries. To improve the estimation quality, we also apply the following methods: 1) we enlarge the moving block size by adding borders around it; 2) if some blocks overlap at the estimated frame, we calculate their mean value as the final image values; 3) if the moving blocks contain no sky texture (e.g. the blocks contain part of shadowband or camera-holding arm), we only move the partial block with sky texture.

We can determine the order of wind field in terms of altitude, which is extremely important, but not considered in [6]. For every frame we obtain the enclosed areas of different cloud layer. With the current wind field speed, we can detect whether there will be any area with overlapping cloud layers in  $g$  minutes later. Once we get the real frame for  $g$  minutes later, we examine the same area to validate the estimation result. The detected wind field for this area would have lower altitude than

---

**Algorithm 2:** MotionVector-Adjustment( $V, th_{cos}, lth_{ratio}, uth_{ratio}$ )

---

**Output:** Adjusted motion vector matrix  $V'$

- 1  $GW F =$  Windfield-Extraction( $V, th_{cos}, lth_{ratio}, uth_{ratio}$ );
- 2 **foreach**  $\vec{v}_{(i,j)}$  of  $V$  **do**
- 3     Let  $LV$  be the local neighborhood motion vectors of  $\vec{v}_{(i,j)}$  ;
- 4      $LWF =$  Windfield-Extraction( $LV, th_{cos}, lth_{ratio}, uth_{ratio}$ );
- 5     Let  $v_{hf}$  be the highest frequency windfield ;
- 6     **foreach**  $\vec{v}_{wf}$  of  $LWF$  **do**
- 7          $sim = \cos(\vec{v}_{wf}, \vec{v}_{hf})$  ;
- 8          $ratio = \|\vec{v}_{wf}\|_2 / \|\vec{v}_{hf}\|_2$  ;
- 9         **if**  $sim \geq th_{cos}$  and  $lth_{ratio} \leq ratio \leq uth_{ratio}$  **then**
- 10              $\vec{v}_{(i,j)}^t$  of  $V' = \vec{v}_{hf}$  ;
- 11             **continue** to step 2 ;
- 12         **end**
- 13     **end**
- 14 **end**

---

the others. For every ten minutes the program re-examine the wind field order. During estimation, we only keep the lowest cloud texture in the future overlapped area of estimated frame, which is different from the single layer cloud estimation.

## VII. SOLAR IRRADIANCE PREDICTION

In this Section, we provide a linear prediction model for real-time solar irradiance forecasting based on our cloud motion estimation and the previous solar irradiance monitoring data. It is expected that the ground radiation reading experiences a deduction in sync with the cloud's motion when it passes the sun location in the sky. Even our model might potentially oversimplify the factors of solar irradiance, our later analysis shows that the model nonetheless performs reasonably well and is even superior to those models relying heavily on external climate data provided by solar site's nearby weather stations. The solar power output estimation is not in the scope of this paper since it is also determined by various factors (such as panel manufacturer, orientation and surface temperature).

We consider the following model for the correlation between TSI image and solar irradiance values. Both data streams are treated as discrete time series, namely  $Rad(t)$  and  $RBR(t)$ , each of which is sampled at two different step sizes: respectively one- and two- minute intervals. In the linear prediction model used by our paper, the value at one time step depends only on the immediate preceding time step:

$$Rad(t) = Rad(t-1) + C \times (RBR(t) - RBR(t-1)). (6)$$

Here  $Rad(t)$  is the time series of pyranometer radiation readings in  $W/m^2$ , and  $(RBR(t) - RBR(t-1))$  defines a time series of the TSI image red-blue ratio (RBR) value change w.r.t the prior step. Our choice to use RBR as an indicator of clouds is validated by the finding that RBR consistently differentiates clear sky v.s. different levels of cloudy sky. The coefficient  $C$  in Equation (6) defines the correlation between cloud and radiation which should be negative.

Based on statistical analysis of over thirty thousands of TSI images with various cloud conditions, we found that the RBR value of cloud pixels consistently falls into the range between 0.6 and 1, while theoretically this range could have been much bigger. A similar threshold (termed as Sunshine Parameter) is made of essential use in [6] for binary characterization of cloudy and clear sky conditions. Comparably, in our model we make no essential use of the precise value as threshold to distinguish clouds from clear backgrounds.

## VIII. EXPERIMENTS

### A. Experiments Setup

Our experiments use four days data from Southern Great Plains Central Facility, Ponca City, Oklahoma (SGP), and two days data from Tropical Western Pacific Darwin Site, Northern Territory, Australia (TWP) collected by ARM program [8], with a wide variety of cloud conditions. We only pay attention from 3 hours before to 3 hours after the local solar culmination time (720 TSI images in 6 hours with every 30 seconds per frame). During this time period, the cloud coverage has higher correlation with actual solar irradiance. June 27th 2010 from SGP and September 11th 2010 from TWP have high cloud coverage, and radically shape-changing cloud with multiple layers. July 12th 2010 from SGP shows single layer cloud but with high sky coverage and mild cloud shape changes. August 6th 2010 from SGP shows multi-layer cloud with motion direction that changes smoothly. October 25th 2010 from TWP has low cloud coverage, but radically shape-changing cloud with multiple layers. August 16th 2010 from SGP has scattered single layer cloud. We did not consider sunny or rainy days because it is straightforward to predict the solar radiation level during these days. Such diverse data selection is intended to study the robustness of our proposed algorithms. Comparably, TWP has very dramatically changing cloud due to the tropical weather, hence experiments are tested on 0.5, 1 and 2 minutes prediction. The experiments on SGP are tested on 1 4, 7 and 10 minutes prediction.

We use the same parameters setting as in [13]. Especially, earth mover's distance (EMD [17]) was used as cloud motion estimation evaluation metric because EMD has very nice properties on revealing image similarity. EMD allows for partial matches in a very natural way which is of importance to deal with image occlusions and clutter. We use the most recent frame as our baseline(*BL*) approach, and compare motion estimation with using only two frames (*2F*) which is equal or better than [6] due to refined gap filling, multi frames using sequential motion vector prediction (*MF*) and *MF* with wind field adjustment (*MF + WF*). We compare our estimated results to the (future) ground truth of TSI images.

### B. Cloud Motion and Solar Irradiation Prediction Results

Motion estimation results on SGP are shown in Figure 6(a) for overall evaluation and Figure 6(b) for each days. Overall our proposed *ML + WF* significantly improves the motion estimation quality from 14.9% to 27.8%. *2F* shows the minimum 14.9% improvement over baseline. On one minute estimation, all prediction models show quite similar results. As we increase the prediction span to ten minutes, our sophisticated *MF* and *MF + WF* models show much

better performance (Figure 6(b)). The result for TWP cloud estimation, confirms that *MF + WF* has better capability to predict cloud motion with radically change shape compared with *2F* model, since our model outperforms not only *BL* with 46.59% (Figure 6(c)), but also the existing typical cloud motion estimate using *2F*. Although *MF + WF* only slightly improves less than 4% based on *MF* on the TWP tests, which is because of the dramatically changing wind-field in TWP site.

We used data on September 11, 2010 data at TWP for illustrating our solar irradiation prediction result. We made two predictions, one and two minutes respectively, with our linear prediction model (LP). We used the binary model in [6] as baseline for comparison, and set time step to be 1 minute. From the TSI's geographical location and date, we can accurately calculate Sun's position on the original TSI image. Then we use the coordinate transformation in the preprocessing algorithm to map Sun's position into the corresponding pixel in the undistorted TSI image. Around this pixel, we choose various sizes for the surrounding block ranging from  $10 \times 10$  to  $50 \times 50$  pixels. Experiments confirmed that a window of such sizes entails the information with a strong correlation to the pyranometer reading changes, while at the same time avoids the random errors from motion estimation algorithm. According to the result, the window size of  $20 \times 20$  is consistently optimal, and therefore selected for subsequent analysis. We setup an upper bound curve for the predicted solar irradiance values. The upper bound changes along with the time during course of six hours. For the experiment, we chose the values from the radiance curve of TWP on September 17, 2010 which was a perfect clear sky all day long. The effect on the predicted real-time irradiance value curve can be seen from the plateaus in plots (a) and (b) of Figure (7). Nevertheless, in the binary model [6] a lower bound of 40% of global horizontal irradiance (GHI) was imposed. Consequently, the model produced noticeable deviations from actual pyranometer radiation readings on the randomly selected day.

We used 10-fold cross validation for our LP model based on the TWP sample. We did not obtain the executable of the baseline binary method, therefore had to skip their cloud motion estimation step, and fed the competing baseline model directly with the ground truth, i.e. the actual TSI cloud image of the next time step. The output of this baseline is actually generated directly by filtering the real TSI data with their threshold method prescribed in [6]. Two thresholds used in the baseline model are called GHI and SP. The root-mean-square error (RMSE) scores of the baseline model under this treatment is much better than that of the actual step-by-step execution. The respective RMSE results are given in Table I. The baseline

TABLE I. RMSE RESULTS COMPARISON.

UCSD binary baseline	1 min LP	2 min LP
206.85	142.08	177.72

[6] is not a real prediction and hence we need not perform cross validation on it. The comparison is then biased towards the binary method, as it also has no propagated error from the motion estimation algorithm. Nevertheless, the advantage of our model is validated by a smaller (better) RMSE value.

Our testings also confirmed a negative correlation between the RBR value within the solar window and the solar irradiance

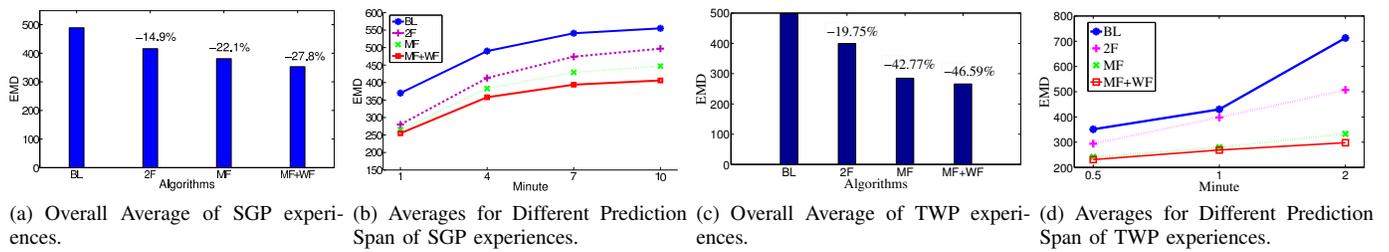


Fig. 6. Comparison of four cloud motion estimation algorithms.

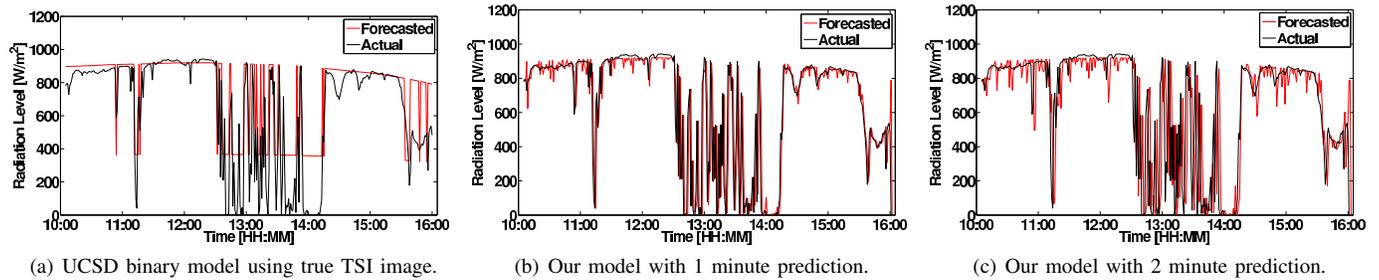


Fig. 7. Comparisons of our radiation prediction models with UCSD model.

value. The experimental values range between  $C = -46.08$  and  $-69.74$  for the 10-fold cross validations for the 1-minute and 2-minute forecasts.

## IX. CONCLUSION

We introduced cloud motion estimation algorithms for short term solar radiation level predictions and addressed special challenges associated with cloud prediction using TSI images and pyranometer solar irradiance readings. In particular, we adopted fast cross correlation method to satisfy the real-time processing requirement and improved the detection accuracy with several refinement techniques. Furthermore, we implemented a sequential motion vector prediction algorithm to deal with cloud shape changes, and proposed wind-field detection and motion vector adjustment algorithms to handle multi-layer clouds. Finally, we provided a prediction pipeline of integrating these algorithms and methods to process the workflow from the image preprocessing step to the final cloud motion estimation and solar irradiance forecast. The experiments demonstrated that our approach significantly outperforms the baseline estimation with more than 28% improvement on average. We expect that our future development implementing multiple TSI devices will both greatly expand the length of the prediction time window and improve the quality of our motion estimation and solar irradiance predictions.

## ACKNOWLEDGMENTS

This research is part of “A Public-Private-Academic Partnership to Advance Solar Power Forecasting”. It is supported in part by DOE grants DE-AC02-98CH10886.

## REFERENCES

- [1] D. I. Barnea and H. F. Silverman. A class of algorithms for fast digital image registration. *IEEE Trans. Computers*, 21:179–186, 1972.
- [2] X. Bresson, S. Esedoglu, P. Vanderheynt, J. P. Thiran, and S. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and vision*, 28(2):151–167, 2007.
- [3] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *ECCV*, 2004.
- [4] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):500–513, 2011.
- [5] T. Chanwimaluang, G. Fan, and S. R. Fransen. Hybrid retinal image registration. *IEEE Trans. Info. Tech. Biomed.*, 10(1):129–142, 2006.
- [6] C. W. Chowa, B. Urquhart, M. Lavea, A. Dominguez, J. Kleissla, J. Shieldsb, and B. Washomc. Intra-hour forecasting with a total sky imager at the uc san diego solar energy testbed. *Solar Energy*, 85(11):2881–2893, 2011.
- [7] S. Cui, Y. Wang, and J. E. Fowler. Mesh-based motion estimation and compensation in the wavelet domain using a redundant transform. *International Conference on Image Processing*, 1:I–693, 2002.
- [8] Atmospheric Radiation Measurement (ARM) Climate Research Facility. Tsi images. <http://www.arm.gov/instruments/tsi>, 2010.
- [9] R. L. Fogt and D. H. Bromwich. Atmospheric moisture and cloud cover characteristics forecast by amps. *Wea. Forecasting*, 23:914–930, 2008.
- [10] P. Glomb. Detection of interest points on 3d data: Extending the harris operator. *Computer Recognition Systems*, 57(3):103–111, 2009.
- [11] T. J. Greenwald, R. Hertenstein, and T. Vukicevic. An all-weather observational operator for radiance data assimilation with mesoscale forecast models. *Monthly Weather Review*, 130:1882–1897, 2002.
- [12] P. Heas and E. Memin. Three-dimensional motion estimation of atmospheric layers from image sequences. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8):2385–2396, 2008.
- [13] H. Huang, S. Yoo, D. Yu, D. Huang, and H. Qin. Cloud motion detection for short term solar power prediction. *ICML 2011 Workshop on Machine Learning for Global Challenges*, 2011.
- [14] J. P. Lewis. Fast normalized cross-correlation. *Vision Interface*, 1995.
- [15] J. Marcello, F. Eugenio, and F. Marques. Cloud motion estimation in seviri image sequences. *IEEE IGARS*, 2009.
- [16] R. R. Rogers. A short course in cloud physics, 2nd edition.
- [17] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover’s distance as a metric for image retrieval. *IJCV*, 40(2):99–121, 2000.
- [18] T. W. Ryan and B. R. Hunt. The prediction of cross-correlation accuracy in digital stereo-pair images. *Electro-optical technology for autonomous vehicles*, pages 121–128, 1980.
- [19] T. Vukicevic, T. Greenwald, M. Zupanski, D. Zupanski, T. Vonder Haar, and A. S. Jones. Mesoscale cloud state estimation from visible and infrared satellite radiances. *Monthly Weather Review*, 132(12):3066–3077, 2004.