Pattern Classification and PSO Optimal Weights Based Sky Images Cloud Motion Speed Calculation Method for Solar PV Power Forecasting

Fei Wang
State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources (North China Electric Power University), Baoding 071003, China; also with Department of Electrical Engineering, North China Electric Power University, Baoding 071003, China; also with Hebei Key Laboratory of Distributed Energy Storage and Micro-grid (North China Electric Power University) Baoding 071003, China
feiwang@ncepu.edu.cn

Miadreza Shafie-khah
C-MAST, University of Beira Interior
6201-001 Covilhã, Portugal
miadreza@ubi.pt

Shuaijie Pang, Zhao Zhen, Kangping Li and Hui Ren
Department of Electrical Engineering
North China Electric Power University
Baoding 071003, China
shuaijiepang@sina.cn
zhenzhao@ncepu.edu.cn
kangpingli@ncepu.edu.cn
hren@ncepu.edu.cn

João P. S. Catalão
INESC TEC and FEUP, 4200-465 Porto, Portugal;
also with C-MAST/UBI, 6201-001 Covilhã, Portugal;
also with INESC-ID, IST-UL, 1049-001 Lisbon, Portugal
catalao@ubi.pt

***Abstract*—The motion of cloud over photovoltaic (PV) power station will directly cause the change of solar irradiance, which indirectly affects the prediction of minute-level PV power, so the tracking of cloud motion is very crucial. In this study, Block-matching algorithm, Optical Flow algorithm and feature matching algorithm are three prevailing methods. However, as a rigid registration method, Block-matching cannot obtain the parameters of cloud deformation or rotation. The accuracy of the optical flow, which is based on the assumption that the image grayscale is not changed, is easily disturbed by noise. When the image texture information is not rich enough, the accuracy of the feature matching will also be reduced. That is, in order to improve their robustness, they must be combined through a certain strategy. Therefore, a pattern classification and PSO optimal weights based sky images cloud motion speed calculation method for solar PV power forecasting (PCPOW) is proposed in this paper. The method consists of two parts. Firstly, we use k-means clustering method and texture features based on Gray-Level Co-occurrence Matrix (GLCM) to classify the clouds. Because texture can adequately reflect image information, compared with other image features, it can better take into account both the macro nature and the fine structure of images. Secondly, for different cloud classes, we build the corresponding combined calculation modeling to obtain cloud motion speed. The Particle Swarm Optimization algorithm is used to give different weights to different methods to adapt to different clouds. The performances of the method are investigated using real data recorded at Yunnan Electric Power Research Institute. Under the measurement of common precision index, the comparisons with various benchmark methods show the effectiveness of the proposed approaches over cloud tracing.**

***Index Terms*--Cloud Motion Speed; Combined Modeling; Optimal Weights; Pattern Classification; Sky Image.**

I. INTRODUCTION

As a significant fashion to utilize the solar energy, the photovoltaic (PV) power has gained burgeoning expansion of late years owing to its merits of no fuel consumption, no pollutant emission, and flexible configuration.

However, PV belongs to intermittent power supplies, and to be more exact, there are randomness and volatility in PV output as it is affected by meteorological factors, i.e. solar irradiance, ambient temperature, moisture, wind velocity and barometric pressure, etc. [1]-[5]. These shortcomings may bring rigorous challenges to power balance, security stability and economic operation of power system [6]-[9]. From the above, we can see that in order to provide a credible understructure for power system scheduling decision-making behavior and ameliorate its capacity to consume intermittent power, it is the time to put forward an accurate and effective prediction scheme of PV. The cloud motion, such as birth, dissipation and deformation, are pivotal elements to the transformation of solar irradiance, thus giving rise to the change of PV output. Therefore, the investigation of cloud motion has turned out to be one of the extremely critical tasks to complete the above-mentioned forecast methodology.

In the related research of cloud motion, most investigators take advantage of satellite images for analysis and processing [10]-[13] in an early phase. However, satellite images are not ideal for the elaboration of regional or low cloud info as a result of low spatial and temporal resolution [14]. Currently, in order to track the movement of the local cloud (especially over the PV power station) more accurately, scholars obtain the speed of cloud by means of ground-based images. However, a full-fledged and effective cloud tracing method has not been proposed up to present.

There are three main ways to calculate the cloud speed: Block-Matching, Optical Flow and feature matching algorithm. The Block-Matching algorithm performs the work by measuring the similarity of the sub blocks between adjacent images. And it is adopted by Chow et al. in [15], Huang in [16] and Peng et al. in [17].

The Optical Flow algorithm is a pixel-level non-rigid registration method, according to the assumption that the gray level of the image remains unchanged. Following original approaches presented in Horn and Schunck (HS) in [18] as well as Lucas and Kanade (LK) in [19], Optical Flow algorithm is used by Wood-Bradley in [20], Chow in [21], and Brox in [22]. As for the feature matching algorithm, this is a broad concept. You need to find corresponding features (e.g. Harris points, SIFT points or SURF points) in adjacent images to complete the task of regional matching. Based on the idea of feature matching, Cheng adopted SIFT algorithm in [23], and F Su used SURF algorithm in [24]. However, as a rigid registration method, Block-Matching cannot obtain the non-rigid motion parameters of cloud, such as rotation and deformation. The accuracy of the optical flow, which is based on the assumption that the image grayscale is not changed, is easily disturbed by noise. For example, in the case of uneven illumination, the computational accuracy is on the low side. As for feature matching, because the definition of a feature point usually requires a lot of texture information, it is poorly matched in regions where the texture information is not rich enough. In short, the above three methods have poor robustness. Because the clouds in the sky images may have various forms, there are above demerits that cannot be overcome when we use the same single method for all kinds of clouds. Furthermore, any simple combined calculation modeling without cloud classification is difficult to achieve good results as well.

In this paper, we propose a pattern classification and PSO optimal weights based sky images cloud motion speed calculation method for solar PV power forecasting. The related works are mainly divided into the following two parts: Firstly, we classify cloud according to the texture feature information of sky images captured by ground-based sky imager. Secondly, for different cloud classes obtained in the previous step, we utilize PSO to optimize the weights of three methods: Block-Matching, Optical Flow and SURF feature matching algorithm, and build the corresponding combined calculation modeling.

In the light of the method proposed, we can choose different calculation strategies (set different weights of three methods in the combined modeling) according to different clouds. In other words, it could improve the weaknesses of the traditional single method in the applicable scope and it is a more universal modeling suitable for most cloud scenes. As for the second chapter, we will give a detailed description of the proposed method "PCPOW". In the third chapter, we will utilize measured data to validate the proposed method and compare the results with those of single common algorithm. Finally, in the fourth chapter, we will summarize the work of the full paper and discuss the future work.

II. THE PCPOW METHOD

The flowchart of the PCPOW is shown in Fig. 1.

A. Pattern Classification of Clouds

For clouds in sky images, we can describe them in terms of brightness, size, shape, spectrum, and texture features, etc.

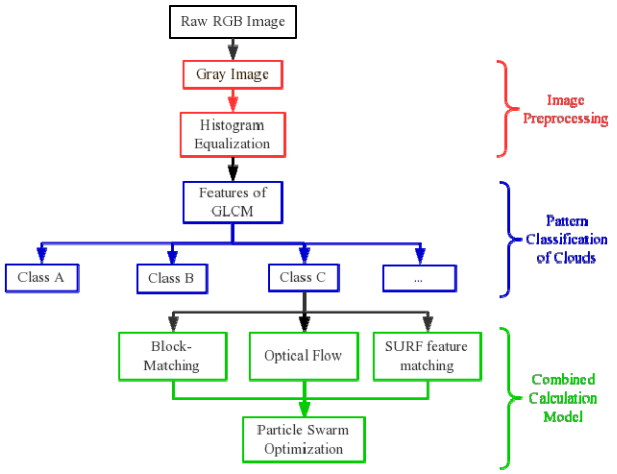

Fig.1 Flowchart of the PCPOW

As a regional feature, texture is a description of the spatial distribution of each pixel in an image.

We can simply understand that texture consists of texture primitives that are repeated in accordance with certain rules or statistical rules. Because texture can adequately reflect image information, compared with other image features, it can better take into account both the macro nature and the fine structure of images. In this paper, we use the texture features based on Gray-Level Co-occurrence Matrix (GLCM) to classify the clouds.

1) Histogram Equalization

Many of the sky image textures are not rich enough, and in order to improve the speed of the program, we have compressed the resolution of original sky images. These will lead to the loss of image information, so we must enhance the image firstly.In this paper, we employ histogram equalization to enhance the image. If the pixels of an image share many gray levels and are evenly distributed, then such images tend to have high contrast and variable gray tones. Histogram equalization is a transformation that can automatically achieve this effect by relying only on the histogram information of the input image. Its basic idea is to widen the gray levels which contain more pixels in the image, and to compress the gray levels which contain fewer pixels in the image. Thus, the dynamic range of the pixel gray value is extended, and the contrast and the change degree of the gray tone hue are improved, and a clearer image is generated.

2) Gray-Level Co-occurrence Matrix

Take points (x, y) and (x + a, y + b) as a point pair in the image. The gray value of the point pair is (i, j), and that is to say, the gray value of the point (x, y) is i, and the gray value of the point (x + a, y + b) is j. Fix a and b, and move the point (x, y) on the whole image, you'll get all sorts of (i, j). Let the gray level of the image be L, then the combination of i and j has a total of L^2 species. In the whole image, the number of occurrences of (i, j) is counted, and then they are normalized to the probability P_{ij}, then the square $\left[P_{ij}\right]_{L*L}$ is the GLCM.

For slowly changing textures (i.e. coarse textures), the GLCM has a larger diagonal value, and the two sides are smaller, that is, GLCM tends to be diagonal distributions; on the contrary, for fast changing textures (i.e. fine textures), it tends to be evenly distributed. Obviously, different (a, b) combinations can get different GLCMs.

3) Features based on Gray-Level Co-occurrence Matrix

Based on GLCM, we select some characteristic quantities to reflect the condition of the matrix, as shown:

Energy:

$$f_1 = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} P_{ij}^2 \tag{1}$$

It reflects the uniformity of the image gray scale distribution and the texture roughness. When the P_{ij}, value distribution is more concentrated, the energy is larger. When the P_{ij} value distribution is more dispersed, the energy is smaller.

Correlation:

$$f_2 = \frac{1}{\sigma_x \sigma_y}\sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-\mu_x)(j-\mu_y)P_{ij} \tag{2}$$

where:

$$\mu_x = \sum_{i=0}^{L-1} i \sum_{j=0}^{L-1} P_{ij} \tag{3}$$

$$\mu_y = \sum_{j=0}^{L-1} j \sum_{i=0}^{L-1} P_{ij} \tag{4}$$

$$\sigma_x^2 = \sum_{i=0}^{L-1}(i-\mu_x)^2 \sum_{j=0}^{L-1} P_{ij} \tag{5}$$

$$\sigma_y^2 = \sum_{j=0}^{L-1}(j-\mu_y)^2 \sum_{i=0}^{L-1} P_{ij} \tag{6}$$

It measures the similarity of the matrix elements in the row and column directions, that is, the local gray correlation in the image is reflected. When the matrix element gap is small, the correlation value is large. When the matrix elements differ greatly, the correlation value is small.

Entropy:

$$f_3 = -\sum_{i=0}^{L-1}\sum_{j=0}^{L-1} P_{ij}\log_2 P_{ij} \tag{7}$$

Entropy is a measure of the amount of information of an image, which represents the complexity of the texture. If the distribution of P_{ij} is relatively uniform, then the entropy is large. If the distribution of P_{ij} is more concentrated, the entropy is smaller.

Contrast:

$$f_4 = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1} |i-j|^2 P_{ij} \tag{8}$$

For coarse textures, contrast is small. For fine textures, contrast is large.

In this paper, we take (a, b) of 4 combinations (1, 0), (-1, 1), (-1, 0) and (-1, 1) to generate 4 GLCMs for each sky image, and the corresponding energy, correlation, entropy and contrast mean values are calculated as the four dimensional texture feature vectors of each image. Then, we take the mean of the feature vectors of two adjacent images as their common feature vectors, and take them as samples of pattern classification.

4) K-means clustering

The k-means clustering method is approached to achieve the classification according to the feature vectors obtained in the previous step. It iterative partitioning minimizes the sum, over all clusters, of the within-cluster sums of point-to-cluster-centroid distances. The squared Euclidean distance is used in this approach [25].

Here, we use the Calinski-Harabasz (CH) criterion to determine the optimal number of clusters, that is, the K value.

$$CH = \frac{\sum_{i=1}^{K} \left\| x_i - \bar{x} \right\|}{\sum_{i=1}^{K}\sum_{p\in C_i} \left\| p - x_i \right\|} \tag{9}$$

Where: p is the sample point in the class C_i, x_i is the clustering center of C, and $\bar{x}$ is the sample mean. Obviously, the larger the value of CH, the more optimized the number of clusters. In this paper, in order to visually represent the change in the value of the criterion, we take CH to the inverse. In this way, we need to select the corresponding K value when the value of criterion is small adequately and the decline trend becomes slow enough.

B. The Three Submethods of Combined Calculation Modeling

We explain briefly the principles of the three sub methods mentioned above firstly, and then introduce the so-called combined calculation modeling.

1) Block-Matching algorithm

The rationale of Block-Matching is that every last image in the frame sequence is subdivided into sub blocks, and afterward, compares all the candidate sub blocks in a given search area of the current frame to one sub block of the previous frame. Owning to the principle of minimum difference of accumulated gray value, the sub block which resembles the previous sub block most is found, namely matching block. In this way, the displacement between the previous block and the matching block is the motion vector of the block.

Usually, one of the following four formulas is used to measure the difference in gray values between the two images:

Correlation coefficient:

$$R_1 = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_1(x_i,y_j)-\overline{f_1}][f_2(x_i,y_j)-\overline{f_2}]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_1(x_i,y_j)-\overline{f_1}]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_2(x_i,y_j)-\overline{f_2}]^2}} \quad (10)$$

$$R_2 = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}f_1(x_i,y_j)f_2(x_i,y_j)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}f_1(x_i,y_j)^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}f_2(x_i,y_j)^2}} \quad (11)$$

Variance:

$$R_3 = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[f_1(x_i,y_j)-f_2(x_i,y_j)]^2 \quad (12)$$

Sum of grey value differences:

$$R_3 = \sum_{i=1}^{M}\sum_{j=1}^{N}|f_1(x_i,y_j)-f_2(x_i,y_j)| \quad (13)$$

where: $f_1(x_i,y_j)$ and $f_2(x_i,y_j)$ are the gray function of two images I_1 and I_2. I_1 is the previous frame image and I_2 is the next frame image. $\overline{f_1}$ and $\overline{f_2}$ are the gray average, $M*N$ is the size of images, that is to say, it is the size of the corresponding gray scale matrixes.

Obviously, when R_1 and R_2 are at their maximum, or R_3 and R_4 are at their minimum, the two images are the most similar.

However, the calculation of the original Block-Matching algorithm is very large, so that it is difficult to meet the requirements of real-time computing. In order to reduce the workload, a Block-Matching Fast Fourier Transform algorithm [26] is widely used. In this method, the digital image is regarded as a sequence of discrete two-dimensional signal field with time, and the computation speed is greatly improved by using the method of signal analysis.

To sum up, Block-Matching is a widely used and very basic method for cloud velocity calculation. In the combined calculation modeling below, we used the corresponding convenient Particle Image Velocimetry toolbox in Matlab to accomplish the mission of cloud speed calculation.

2) Optical Flow algorithm

The optical flow is the "instantaneous velocity (u, v)" of the moving object in the observed image plane. As for Optical Flow algorithm, its essence is to establish the optical flow constraint equation in line with the truth of image intensity conservation. By solving the equation above, the velocity parameters can be acquired.

There are three premise hypotheses of optical flow:

Hypothesis 1: The gray levels of the corresponding pixels in adjacent images are constant.

Hypothesis 2: The displacement of the target in the adjacent image is relatively small.

Hypothesis 3: A pixel has the same displacement as the pixel in the neighborhood.

The first step: according to the hypothesis 1, we can get that:

$$f(x,y,t) = f(x+dx,y+dy,t+dt) \quad (14)$$

where: $f(x,y,t)$ is the gray function of the image. And (x,y) represent the pixel coordinates, and t represents the time. Take Taylor decomposition on the right side of the formula, we can get that:

$$f(x+dx,y+dy,t+dt) = f(x,y,t) + \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial t}dt + R_1(x,y,t) \quad (15)$$

where:

$$R_1(x,y,t) = o(\sqrt{(dx)^2+(dy)^2+(dt)^2}) \quad (16)$$

The second step: based on the assumption that 2, the remainder can be neglected, we can get that:

$$f(x+dx,y+dy,t+dt) = f(x,y,t) + \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial t}dt \quad (17)$$

Let $f_x = \frac{\partial f}{\partial x}$, $f_y = \frac{\partial f}{\partial y}$, $f_t = \frac{\partial f}{\partial t}$, $u = \frac{dx}{dt}$ and $v = \frac{dy}{dt}$, we can get that:

$$f_x u + f_y v = -f_t \quad (18)$$

Written in matrix form:

$$\begin{bmatrix} f_x & f_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} = -f_t \quad (19)$$

However, two unknown quantities (u, v) cannot be obtained by one equation.

The third step: according to hypothesis 3, if the displacement of n points in the neighborhood is the same, then the following n (n>2) equations are obtained:

$$\begin{bmatrix} f_{x1} & f_{y1} \\ f_{x2} & f_{y2} \\ \vdots & \vdots \\ f_{xn} & f_{yn} \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} f_{t1} \\ f_{t2} \\ \vdots \\ f_{tn} \end{bmatrix} \quad (20)$$

The fourth step: using the least squares method to solve the above linear equations, we can get (u, v):

Let $\begin{bmatrix} f_{x1} & f_{y1} \\ f_{x2} & f_{y2} \\ \vdots & \vdots \\ f_{xn} & f_{yn} \end{bmatrix}=A$, $\begin{bmatrix} u \\ v \end{bmatrix}=x$, and $-\begin{bmatrix} f_{t1} \\ f_{t2} \\ \vdots \\ f_{tn} \end{bmatrix}=b$, and the above formula is simplified as :

$$Ax=b \tag{21}$$

Let the optimization index be the sum of the least squares of errors. That is, the variable x is obtained when $\|Ax\text{-}b\|^2$ reaches the minimum value.

In the combined calculation modeling below, we use LK optical flow method to accomplish the mission of cloud speed calculation.

3) SURF feature matching algorithm

When it comes to feature matching, you need to find the feature points that correspond to each other in the two successive images, and then calculate the distance between them. Common feature extraction methods include corner features (such as Harris operator [27]), line features (image edge detection), local area characteristics (speckle features) and invariant features (scale invariant features) and so on.

In this paper, we select the SURF feature operator [28] for image matching. It can extract the feature points better in the case of illumination change and image change, and the scale invariance is better than Harris operator, the time complexity is better than the SIFT operator. The construction of the SURF operator is very complicated, and we only make a brief explanation.

The first step: detect and locate feature points.

SURF uses Hessian matrix $H(x,\sigma)$ to detect feature points:

$$H(x,\sigma)=\begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \tag{22}$$

Where: x is the characteristic point coordinate, σ is the scale, $L_{xx}(x,\sigma)$ is the convolution of the image with the Gaussian second order differential $\frac{\partial^2}{\partial x^2}g(\sigma)$ ($g(\sigma)$ is the Gaussian function, the other meaning is similar).

In order to reduce the running time of the algorithm, the Gaussian template is simplified as a rectangular region box filter, and D_{xx} , D_{yy} and D_{xy} are used to denote the convolution of the box and the image. The determinant of the Hessian matrix can be approximated as follows:

$$\Delta\ (H)=D_{xx}D_{yy}-(0.9D_{xy})^2 \tag{23}$$

The scale invariance of the SURF algorithm mainly depends on finding the feature points at different scales. The scale space is divided by the Octaves, and each group represents the response graph of the incremental filtering template and the image convolution.

The first set of scale spaces are 9*9, 15*15, 21*21, 27*27; the other groups increase similar, but the increments are doubled, 6, 12 and 24, respectively. In order to cover all the scales, there is overlap between the group and the group.

In order to locate the feature points, the extreme suppression method in the 3D space is used to find the extreme points. That is, the Hessian matrix feature points are compared with the other 26 values of the 3*3*3 scale space centered on that point, and only if the point is larger or smaller than the other values, it is the local extreme point. Then, interpolation is made in the scale space and the image space to obtain the final feature point location and the scale value.

The second step: generate the feature point description operator.

In order to make the feature have better rotation invariance, it is necessary to assign the main direction to each feature point.

Firstly, in the circular area with the feature point as the center and the scale of 6 times the radius, the Haar wavelet response of each point is calculated.

Then, add all Haar wavelet dx and dy in the range of 60 degrees to form a new vector (m_w,θ_w) :

$$m_w=\sum dx+\sum dy \tag{24}$$

$$\theta_w=\arctan\left[\frac{\sum dx}{\sum dy}\right] \tag{25}$$

The whole circular area is traversed by 0.2rad in length and the longest vector represents the main direction of the feature point.

Afterward, take the feature point as the center, rotate the coordinate axis to the main direction, select the square area of the 20s*20s (s is the scale of the feature points), and divide it into 4*4, i.e. 16 square sub windows, each side window is 5s.

After that, the Gaussian weighted dx and dy along the main direction and perpendicular to the main direction are obtained using a Haar wavelet template with a scale of 2s.

Finally, in each square, the responses of the Haar wavelet in the x and y directions are summed (the x direction is perpendicular to the main direction and the y direction is parallel to the main direction) to form a four dimensional vector:

$$v=\left[\sum dx \quad \sum|dx| \quad \sum dy \quad \sum|dy|\right] \tag{26}$$

In addition, we normalize it to form 16*4, i.e. 64 dimensional SURF description operator. In addition, Haar wavelet response itself has brightness invariance, and the contrast invariance can be achieved by normalization of feature vectors. In general, it improves the anti-interference ability.

C. Combined Calculation Modeling

Particle Swarm Optimization (PSO) is similar to Genetic Algorithm (GA) and is also an iterative optimization algorithm. The system is initialized as a set of random solutions, and the optimal values are searched iteratively. Compared with GA, it is easier to achieve and do not need to adjust too many parameters [29]-[30]. At present, it has been widely used in function optimization, neural network training, fuzzy system control and other applications of GA. In this paper, we use PSO to assign different weights to each method in the combined calculation modeling on the basis of diverse cloud classes. In other words, the combined calculation modeling is built. As shown in (27):

$$V_{\text{Cloud}} = aV_{BM} + bV_{OF} + cV_{SURF} \tag{27}$$

where V_{Cloud} indicates the final cloud motion speed, V_{BM}, V_{OF} and V_{SURF} indicate the cloud motion speed calculated by Block-Matching, Optical Flow, and SURF feature matching algorithm respectively, a, b and c are weights obtained by PSO and they vary with the cloud classes. Here, we use the correlation coefficient R as the optimization function, and the reciprocal of R as the fitness function of the particles in the PSO.

$$R = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_1(x_i,y_j)-\overline{f_1}][f_2(x_i,y_j)-\overline{f_2}]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_1(x_i,y_j)-\overline{f_1}]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[f_2(x_i,y_j)-\overline{f_2}]^2}} \tag{28}$$

Mind: I_1 here is translated from the initial previous frame image based on V_{Cloud} and I_2 remains the initial next frame image. Obviously, the range of R is [0, 1] and the larger the value of R, the more similar the two gray images (that is to say V_{Cloud} is more accurate). In other words, the smaller the fitness of the particle, the better the particle is.

III. CASE STUDY

A. Data description

We use sky images provided by Yunnan Electric Power Research Institute (25°09'53.35"N and 102°39'32"E) as the experimental sample. These photos were taken in August 2015 and at intervals of 1 minute. The resolution of these images is 256*256. Among them, 300 images were selected as training samples and 200 images as testing samples.

B. Results and Comparison with Other Classical Algorithms

1) Pattern Classification of Clouds

The value of the criterion changes as the number of clusters changes as shown in Fig. 2.

Obviously, when K = 10, the value of criterion is small adequately and the decline trend becomes slow enough. That is to say, the K-means clustering method is used to divide the images into 10 categories.

2) Combined Calculation Modeling

In this section, we use PSO to optimize the three weights a, b and c. The population size is set to 100 and the number of evolution times is also set to 100. The trend of optimal fitness with the number of evolution times in the process of weight optimization for a pair of continuous images in a certain class is shown in Fig. 3.

Finally, we obtain the corresponding 10 sets of weights a, b, and c. By the way, each set of weights here is the average value of the weights obtained by the PSO optimization of all the images in the corresponding class. That is to say, for different types of sky images, we have established 10 corresponding models for cloud velocity calculation.

Next, we will test the proposed modeling in this paper using test images and compare the results with those of the three traditional methods.

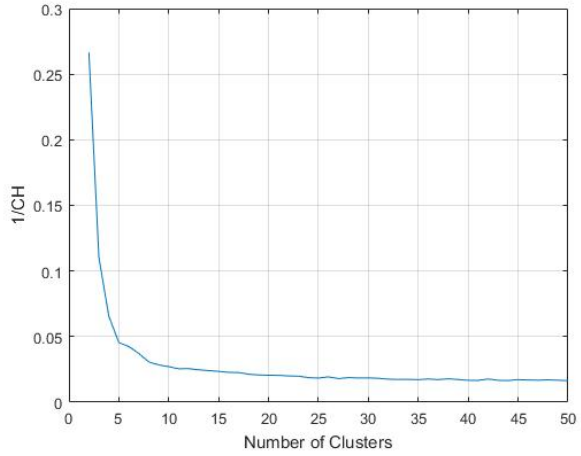

Fig.2 Change trend graph

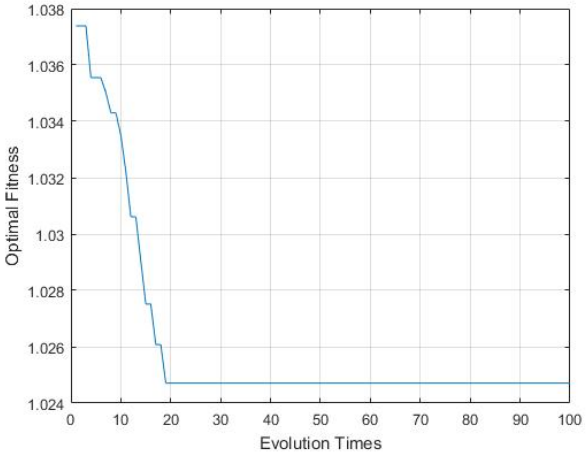

Fig.3 Change trend graph

Of course, for new image sequence, we need to extract the feature vectors described above and calculate the Euclidean distance between them and the above 10 clustering centers to classify images and then use the corresponding computational modeling for cloud velocity calculation.

In the same way, we still use the correlation coefficient R to evaluate the accuracy of the algorithm, as shown in the following Table I and Fig. 4.

It can be seen that the proposed combined calculation modeling in this paper has higher accuracy than the other three traditional methods.

Especially, the calculation precision of SURF feature matching algorithm and Optical Flow algorithm are very low in class 6 and in class 9 separately, but our method can still maintain a relatively high accuracy. Then, we choose several extreme sky images to specify the limitations of the three single methods.

The sky images in Fig. 5 belong to the class 6. In the images, the most similar SURF feature points are marked with green circles.

TABLE I
ALGORITHM COMPARISON

Class	R			
	PCPOW	*Block-Matching algorithm*	*Optical Flow algorithm*	*SURF feature matching*
1	0.9491	0.9441	0.9302	0.8818
2	0.8993	0.8578	0.898	0.7571
3	0.9699	0.9696	0.938	0.8561
4	0.9671	0.9573	0.9466	0.9334
5	0.9694	0.9579	0.9595	0.9592
6	0.8511	0.673	0.7047	0.5659
7	0.9322	0.8923	0.9102	0.9192
8	0.9457	0.9455	0.9425	0.9404
9	0.8771	0.7712	0.6288	0.7291
10	0.8892	0.8325	0.8693	0.8552

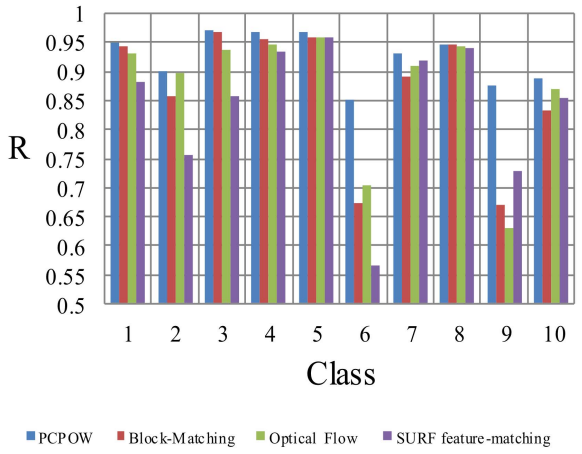

Fig.4 Algorithm comparison

Obviously, the result of the SURF feature matching algorithm indicates that the cloud moves toward the lower left, but in fact the cloud moves toward the lower right. In other words, the two SURF feature points have been mismatched.

Because the texture information of the image is not rich enough, the texture features of each area have little difference, which results in no obvious difference in the 64 dimensional SURF description operator, so the mismatch phenomenon is easy to occur. The sky images shown in Fig. 6 belong to the class 9 and the following pictures show the displacement vector of each pixel in the images. Obviously, the displacement distribution of pixels in the graph is discrete. In other words, the pixels of cloud lack a "common" displacement vector, so when we take the mean value of the displacements of these pixels as the displacement of the cloud, the accuracy of the calculation results will be reduced.

Because the brightness of the corresponding positions of the two images is generally different, this leads to the "Hypothesis 1" is no longer satisfied and the calculation error is produced.

The sky images shown in Fig. 8 also belong to the class 9. Based on the process of Block-Matching algorithm, we divide the image into many sub blocks, then find the most similar blocks according to the principle of gray similarity. Obviously, the sub blocks marked by the red box in the two images are identified as the most similar blocks by the algorithm. In other words, due to the strong deformation of the cloud, the situation of mismatching occurs.

Next, in order to prove the validity of "pattern classification" in this method, we take the average of the corresponding 10 sets of weights a, b, and c obtained above as the common weight of 10 classes of samples. Under the two sets of weight coefficients, the accuracy of the r calculation is shown in Table II.

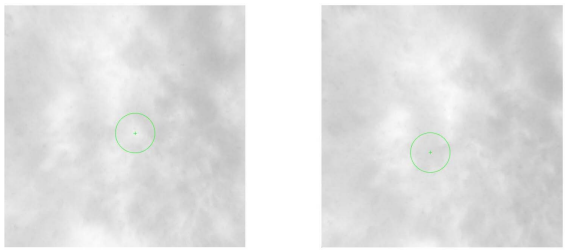

Fig.5 Sky images with flat texture

Fig.6 Sky images with uneven illumination

TABLE II
COMPARISON OF CALCULATION PRECISION

Class	R	
	The different weight coefficients	*The same weight coefficients*
1	0.9491	0.4494
2	0.8993	0.5136
3	0.9699	0.5264
4	0.9671	0.597
5	0.9694	0.9431
6	0.8511	0.8389
7	0.9322	0.8529
8	0.9457	0.8776
9	0.8771	0.8519
10	0.8892	0.887

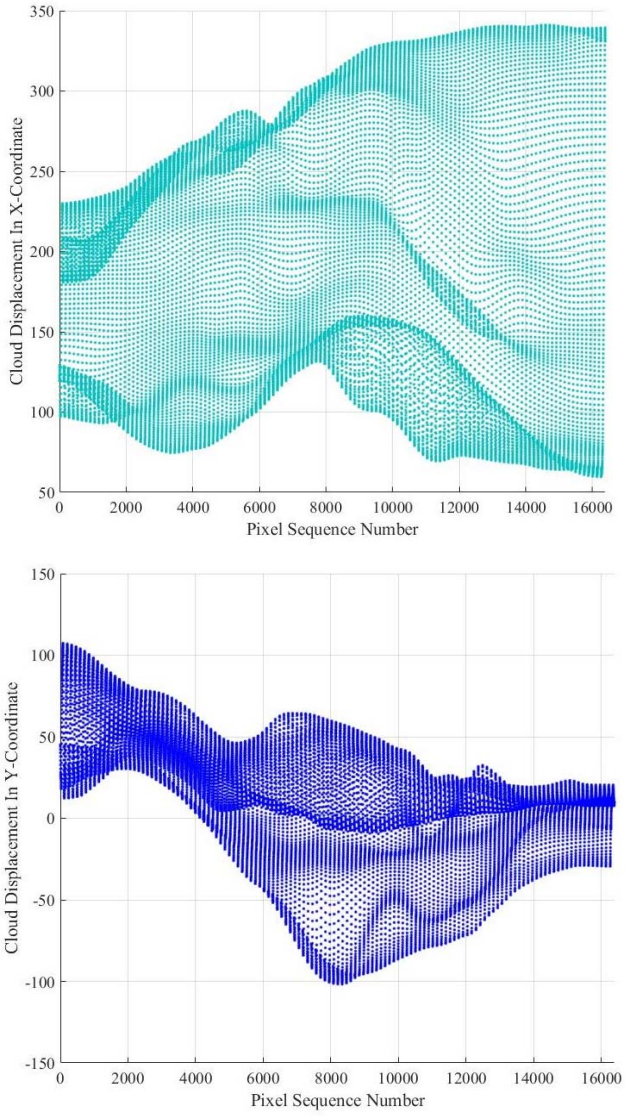

Fig.7 Displacement vector of pixels in the direction of X and Y

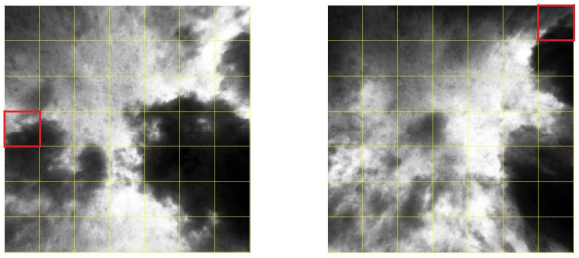

Fig.8 Sky images with strong deformation

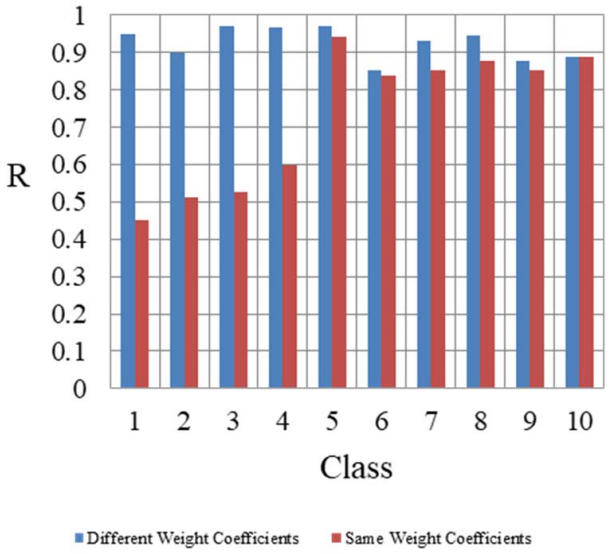

Fig.9 Comparison of calculation precision

It can be seen that if different computing strategies are not adopted for different classes of sky images, the overall accuracy of the testing samples will be greatly reduced. This shows that sky images have different classes and we cannot obtain high precision calculation results by a single computing strategy.

IV. CONCLUSION

As the cloud in sky images has different motion patterns and the existing cloud speed calculation method has poor robustness, it is necessary for us to classify the sky images, and then establish corresponding combination optimization models. In this paper, through the analysis of the three current mainstream methods (Block-Matching, Optical Flow and SURF feature matching algorithm), a pattern classification and PSO optimal weight based sky images cloud motion speed calculation method was proposed for solar PV power forecasting (named PCPOW). Actual sky image data has been used to verify the e□ectiveness of the proposed method. By case study and comparing with the other three traditional methods, the results proved that the method could overcome the weaknesses of the traditional single method in the applicable scope and it provided a more universal modeling suitable for most cloud scenes.

In a future study, attention will be given to the preprocessing of sky image, not only to guarantee the operation speed, but also to ensure that the image information is not lost too much. In addition, how to extract features so as to achieve a more efficient classification of images is also a future research focus.

ACKNOWLEDGMENT

This work was supported by the National Key R&D Program of China (2018YFB0904200), the National Natural Science Foundation of China (51577067), the Beijing Natural Science Foundation of China (3162033), the Hebei Natural Science Foundation of China (E2015502060), the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources (LAPS18008), the Headquarters Science and Technology Project of State Grid Corporation of China (SGCC)(NY7116021), the Open Fund of State Key Laboratory of Operation and Control of Renewable Energy & Storage Systems (China Electric Power Research Institute) (5242001600FB), the Fundamental Research Funds for the Central Universities (2018QN077). J.P.S. Catalão acknowledges the support by FEDER funds through COMPETE 2020 and by Portuguese funds through FCT, under Projects SAICT-PAC/0004/2015 - POCI-01-0145-FEDER-016434, POCI-01-0145-FEDER-006961, UID/EEA/50014/2013, UID/CEC/50021/2013, UID/EMS/00151/2013, and 02/SAICT/2017 - POCI-01-0145-FEDER-029803, and also funding from the EU 7th Framework Programme FP7/2007-2013 under GA no. 309048.

REFERENCES

[1] F. Wang, Z. Zhen, Z. Mi, H. Sun, S. Su, and G. Yang, "Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting," *Energy Build.*, vol. 86, pp. 427–438, Feb. 2015, DOI: 10.1016/j.enbuild.2014.10.002

[2] F. Wang, Z. Zhen, B. Wang and Z. Mi, "Comparative Study on KNN and SVM Based Weather Classification Models for Day Ahead Short Term Solar PV Power Forecasting," *Appl. Sci.*, 2018, 8(1):28. doi.org/10.3390/app8010028

[3] F. Wang, Z. Mi, S. Su, and H. Zhao, "Short-Term Solar Irradiance Forecasting Model Based on Artificial Neural Network Using Statistical Feature Parameters," *Energies*, vol. 5, pp. 1355–1370, Dec. 2012, DOI: 10.3390/en5051355

[4] M. Rana, I. Koprinska, and V. G. Agelidis, "Univariate and multivariate methods for very short-term solar photovoltaic power forecasting," *Energy Convers. Manag.*, vol. 121, pp. 380–390, 2016, DOI: 10.1016/j.enconman.2016.05.025

[5] F. Wang, H. Xu, T. Xu, K. Li, M. Shafie-khah, and J. P.S. Catalão, "The values of market-based demand response on improving power system reliability under extreme circumstances," *Appl. Energy*, vol. 193, pp. 220–231, May. 2017, DOI: 10.1016/j.apenergy.2017.01.103

[6] Y. Sun, F. Wang, B. Wang, Q. Chen, N.A. Engerer and Z. Mi, "Correlation feature selection and mutual information theory based quantitative research on meteorological impact factors of module temperature for solar photovoltaic systems," *Energies*, 2017, 10(1): 7. doi.org/10.3390/en10010007

[7] Q. Chen, F. Wang, B. Hodge, J. Zhang, Z. Li, M. Shafie-Khah, J.P.S Catalão, "Dynamic Price Vector Formation Model Based Automatic Demand Response Strategy for PV-assisted EV Charging Station," *IEEE Trans*. Smart Grid, vol. PP, p. 1, Apr. 2017, DOI: 10.1109/TSG.2017.2693121

[8] F. Wang, L. Zhou, H. Ren, X. Liu, S. Talari, M. Shafie-khah and J. P.S. Catalão, "Multi-Objective Optimization Model of Source-Load-Storage Synergetic Dispatch for a Building Energy Management System Based on TOU Price Demand Response," *IEEE Trans. Industry Applications*, 2018, 54(2):1017-1028. doi: 10.1109/TIA.2017.2781639

[9] F. Wang, Z. Zhen, C. Liu, Z. Mi, B. Hodge, M. Shafie-khah, and J. P. S. Catalão, "Image phase shift invariance based cloud motion displacement vector calculation method for ultra-short-term solar PV power forecasting," Energy Convers. Manag., vol. 157, pp. 123-135, Jan. 2018, DOI:10.1016/j.enconman.2017.11.080

[10] D. Prasad Mukherjee and S. T. Acton, "Cloud tracking by scale space classification," *Geosci. Remote Sensing, IEEE Trans.*, vol. 40, pp. 405–415, Mar. 2002, DOI: 10.1109/36.992803

[11] R. Stuhlmann, M. Rieland, and E. Raschke, "An Improvement of the IGMK Model to Derive Total and Diffuse Solar Radiation at the Surface from Satellite Data," *J. Appl. Meteorol.*, vol. 29, Jul. 1990, DOI: 10.1175/1520-0450(1990)029<0586:AIOTIM>2.0.CO;2

[12] H. Escrig et al., "Cloud detection, classification and motion estimation using geostationary satellite imagery for cloud cover forecast," *Energy*, vol. 55, no. Supplement C, pp. 853–859, 2013, DOI: 10.1016/j.energy.2013.01.054

[13] Z. Peng, S. Yoo, D. Yu, and D. Huang, "Solar irradiance forecast system based on geostationary satellite," *in 2013 IEEE Conference Smart Grid Communications, SmartGridComm 2013*, pp. 708–713.

[14] Z. Peng, D. Yu, D. Huang, J. Heiser, and P. Kalb, "A hybrid approach to estimate the complex motions of clouds in sky images," *Sol. Energy*, vol. 138, no. Supplement C, pp. 10–25, 2016, DOI: 10.1016/j.solener.2016.09.002

[15] C. W. Chow et al., "Intra-hour forecasting with a total sky imager at the UC San Diego solar energy testbed," *Sol. Energy*, vol. 85, no. 11, pp. 2881–2893, 2011, DOI: 10.1016/j.solener.2011.08.025

[16] H. Huang et al., "Cloud motion estimation for short term solar irradiation prediction," *in 2013 IEEE International Conference on Smart Grid Communications, SmartGridComm 2013*, pp. 696–701.

[17] Z. Peng, D. Yu, D. Huang, J. Heiser, S. Yoo, and P. Kalb, "3D cloud detection and tracking system for solar forecast using multiple sky imagers," *Sol. Energy*, vol. 118, pp. 496–519, Aug. 2015, DOI: 10.1145/2554850.2554913

[18] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, no. 1, pp. 185–203, 1981, DOI: 10.1016/0004-3702(81)90024-2

[19] B. D. Lucas and T. Kanade, "An iterative technique of image registration and its application to stereo," 1981.

[20] P. Wood-Bradley, J. Zapata, and J. Pye, "Cloud tracking with optical flow for short-tem solar forecasting," *in 2012 IEEE International Conference on the 50th Conference of the Australian Solar Energy Society*, pp. 55-57

[21] C. W. Chow, S. Belongie, and J. Kleissl, "Cloud motion and stability estimation for intra-hour solar forecasting," *Sol. Energy*, vol. 115, no. Supplement C, pp. 645–655, 2015, DOI: 10.1016/j.solener.2015.03.030

[22] T. Brox and J. Malik, "Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, pp. 500–513, Mar. 2011, DOI: 10.1109/TPAMI.2010.143

[23] H.-Y. Cheng and C.-C. Yu, "Solar irradiance now-casting with ramp-down event prediction via enhanced cloud detection and tracking," *in 2016 IEEE International Conference on Multimedia and Expo*, pp. 11-17.

[24] F. Su, W. Jiang, J. Zhang, H. Wang, and M. Zhang, "A local features-based approach to all-sky image prediction," *IBM J. Res. Dev.*, vol. 59, p. 6:1-6:10, May 2015, DOI: 10.1147/JRD.2015.2397772

[25] S. Su et al., "A classified irradiance forecast approach for solar PV prediction based on wavelet decomposition," *in 2016 International Conference on North American Power Symposium*, pp. 1-5.

[26] C. Willert and M. Gharib, "Digital Particle Image Velocimetry," *Exp. Fluids*, vol. 10, pp. 181–193, Jan. 1991, DOI: 10.1007/BF00190388

[27] B. Yu, L. Wang, and Z. Niu, "A novel algorithm in buildings/shadow detection based on Harris detector," *Opt. - Int. J. Light Electron Opt.*, vol. 125, no. 2, pp. 741–744, 2014, DOI: 10.1016/j.ijleo.2013.07.043

[28] W.-L. Zhao and C.-W. Ngo, "Flip-Invariant SIFT for Copy and Object Detection," *IEEE Trans. Image Process*, vol. 22, Oct. 2012, DOI: 10.1109/TIP.2012.2226043

[29] F. Wang, L. Zhou, H. Ren and X. Liu, "Search Improvement Process-Chaotic Optimization-Particle Swarm Optimization-Elite Retention Strategy and Improved Combined Cooling-Heating-Power Strategy Based Two-Time Scale Multi-Objective Optimization Model for Stand-Alone Microgrid Operation," *Energies*, 2017, 10(12):1936. doi.org/10.3390/en10121936

[30] F. Wang, L. Zhou, B. Wang, Z. Wang, M. Shafie-khah, and J. P. S. Catalão, "Modified Chaos Particle Swarm Optimization-Based Optimized Operation Model for Stand-Alone CCHP Microgrid," *Appl. Sci.*, vol. 7, p. 754, Jul. 2017, DOI: 10.3390/app7080754