Outdoor Shadow Estimating Using Multiclass Geometric Decomposition Based on BLS

Zhihua Chen, Ting Gao, Bin Sheng⬤, Ping Li⬤, and C. L. Philip Chen⬤, *Fellow, IEEE*

Abstract—Illumination is a significant component of an image, and illumination estimation of an outdoor scene from given images is still challenging yet it has wide applications. Most of the traditional illumination estimating methods require prior knowledge or fixed objects within the scene, which makes them often limited by the scene of a given image. We propose an optimization approach that integrates the multiclass cues of the image(s) [a main input image and optional auxiliary input image(s)]. First, Sun visibility is estimated by the efficient broad learning system. And then for the scene with visible Sun, we classify the information in the image by the proposed classification algorithm, which combines the geometric information and shadow information to make the most of the information. And we apply a respective algorithm for every class to estimate the illumination parameters. Finally, our approach integrates all of the estimating results by the Markov random field. We make full use of the cues in the given image instead of an extra requirement for the scene, and the qualitative results are presented and show that our approach outperformed other methods with similar conditions.

Index Terms—Broad learning system (BLS), illumination estimating, Markov random field (MRF), multiclass integrating, shadow synthesis.

I. Introduction

THE SHADOW of the virtual objects in a real scene is one of the key variables, which has to be resolved in the process of augmented reality, image synthesis, object recognition, and so on [1]–[7]. The visual sense of authenticity of an observer to a great extent depends on the accuracy of the shadow synthesis for the virtual object. The shadow synthesis, which must be based on the illumination parameters of the scene, could be more accurate if the illumination condition is precisely estimated. Currently, illumination is a key variable that has to be untangled in the image understanding, recognition, detection, and computer vision fields [8]–[13]. As more and more works have been done in visual problems about illumination [14]–[18], the drawbacks of the traditional illumination estimating algorithms gradually restrict the range of their applications. Many of the existing algorithms are performed in the laboratory setting or indoor scene to simplify the problem [19]. Illumination estimation of an outdoor scene is much more difficult than an indoor scene because of the constantly changing position of the Sun, weather conditions, and the uncertainty of environmental elements.

In illumination estimation [20]–[22], specific objects within the scene are required, such as mirror spheres or prior knowledge. Those methods are limited because there is a little possibility to include needed objects, which are unusual in the real world. As the Sun is the main light source in the outdoor scene, Sun visibility estimating is necessary before illumination condition estimating [23]. While learning-based methods have achieved outstanding results in the field of image processing [24]–[30], we decided to substitute traditional complex algorithms by learning-based methods to achieve part of functions that could be more efficient by learning-based methods. To avoid the unbearable training time of deep learning, we adopt the broad learning system (BLS) in our approach, which is established as a flat network [31]. BLS can greatly reduce the time required for the training process while maintaining the accuracy of the classification. Furthermore, the retraining process is not necessary when the nodes in the system are increased.

Our main idea is to fully utilize the information in given image(s) to estimate the illumination condition. Instead of relying on a specific (or fixed) object, we pave a path for estimating the illumination of an arbitrary given scene as long as a human can. We determine the Sun visibility quickly based on the efficient BLS, which could be expanded on width to establish more complex systems. And for the complex scene, we

Z. Chen and T. Gao are with the Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China (e-mail: czh@ecust.edu.cn).

B. Sheng is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: shengbin@sjtu.edu.cn).

P. Li is with the Faculty of Information Technology, Macau University of Science and Technology, Macau 999078, China (e-mail: pli@must.edu.mo).

C. L. P. Chen is with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau 999078, China, also with the College of Navigation, Dalian Maritime University, Dalian 116026, China, and also with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: philip.chen@ieee.org).

Fig. 1. Synthesis result of our approach. (a) Main input image is necessary, where we will insert our virtual objects. (b) Optional auxiliary input images have the same scene with (a). (c) Synthesis shadow calculated by our method is present in the final image.

have to extract more features from the input image to improve the estimating accuracy. Based on the incremental BLS, we can extract more features from the input image to improve the accuracy of the Sun visibility determination. After the Sun visibility determination, we propose an information classification algorithm based on the geometric structure and the shadow area [we name the algorithm information classification based on geometry and shadow (ICGS)]. The useful information is divided into four categories by ICGS, and the pixels in the same class could estimate the shadow by the same algorithm (Section IV). Finally, we improve the robustness of the shadow synthesis algorithm by the Markov random field (MRF) model for combining all kinds of illumination estimating results from different information cliques (Section V). The estimating accuracy could be improved by optional auxiliary input images, the information extracted in the auxiliary is converted by the image stitching algorithm.

Generally, an optimization algorithm is proposed to integrate multiple images of the same scene and multiclass information in the images. As common sense, we assume that the main light source in the outdoor scene is sunlight, and the position of the Sun determines the illumination condition mostly. The synthesis result of our approach is shown in Fig. 1. The necessary main input is shown in Fig. 1(a), and the synthetic shadow is inserted in the necessary input image [Fig. 1(c)]. The optional auxiliary input images [Fig. 1(b)] have (or partly have) the same scene with Fig. 1(a) and the number of auxiliary input images could be increased for an accurate estimation in the complex scene. To summarize, the three main contributions of this paper are as follows.

1) BLS is used to improve the accuracy of Sun visibility and obtain illumination parameters from flake shadow. We could quickly and precisely determine whether the Sun is visible or not in the general scene by basic BLS. Based on the special structure, incremental BLS could be established without the retraining process quickly to determine the Sun visibility of the complex scene. And the flake shadow is used to estimate the illumination parameters by BLS too.

2) Multiclass geometric decomposition makes full use of cues in given image(s). An information classification algorithm ICGS is proposed, which classifies the information about the light condition in an image based on geometric structure and shadow detection. Especially, the shadow on the ground is classified into two kinds: 1) long shadow and 2) the others (flake shadow).

3) Optional auxiliary input images are adopted to improve the estimating accuracy. While the illumination information may be limited by only one input image, we allow optional auxiliary image(s) in our framework to improve the estimation accuracy. The illumination of the necessary input is estimated by the information both in the necessary input image and the optional auxiliary image(s), that coordinate is castigated by image stitching. The cues from different images are integrated by MRF to improve the robustness of the estimating results.

II. RELATED WORK

A. Broad Learning System

Deep neural networks have achieved huge success in many fields [32]–[34]. The performance of a deep neural network is sensitive to its parameters, such as the learning rate. And the long training time of deep neural networks is unbearable for users. To eliminate the drawback of deep neural networks, the random vector functional-link neural network (RVFLNN) is proposed. Compared with the deep neural network, RVFLNN makes the hidden layers move down as input nodes so that RVFLNN can contact the input nodes and output nodes directly. With the growth of data size and data dimensions, dimension reduction and feature extraction are necessary. BLS is designed based on the inspiration of RVFLNN. Furthermore, BLS can effectively and efficiently update the system because the new weights could be calculated without a retraining process. BLS is established as a flat network, and the broad expansion could be in both the feature nodes and the enhancement nodes [31], [35], [36]. The classification experiments on popular MNIST and MORB data have been done in [31], which shows that BLS outperforms deep structure networks.

B. Sky Physical Model

Lalonde *et al.* [23], [37] estimated the position of the Sun relative to the camera based on the sky physical model. After assuming that the sky is clear, they disperse the parameter space of Sun's zenith angle and azimuth angle, and then match the discretized parameters by the sky physical model. The sky physical model estimates the position of the Sun based on the brightness and position of the sky pixels. Based on the relationship between the position of the Sun and the brightness distribution of sky pixels, the Sun's position, which is represented by the zenith angle and the azimuth angle, can be calculated. They take the sky pixels' brightness and its position into the function, and the camera parameters are estimated from the image. From a large number of sky pixels, the physical model could estimate the position of the Sun accurately. However, this method will be invalid if there are only a few sky pixels (no sky pixels) in the image.

C. Geometric Structure Estimating

Geometric structure is a significant composition of an image, and the process of estimating the geometric structure is still challenging. Hoiem *et al.* [38] predicted the geometric information from a single image. A general geometric structure of the given image can be obtained by [38], and the different parts are labeled by different signs. Hoiem *et al.* [38] can detect the different orientations in the image with respect to the camera.

D. Shadow Detection

Shadow area detection is a challenging topic when the shadow is not well-grounded or the shadow and ground are similar. While more and more shadow detection methods are proposed [39], the new feature found in the frequency-domain could help to detect the shadow accurately [40] in the outdoor scene. They calculate the spectrum ratios under various Sun angles and further. While the existing shadow is a very important cue for the illumination estimation, we have to accurately detect the shadow. In [41], the problem of shadow detection is addressed from the learning-based perspective. They recognize the shadow in the grayscale nature images based on both variant and invariant shadow information from illumination, texture, and odd-order derivative characteristics. The challenge of shadow detection is increased because of the lost color cues and the complex scene in nature images. The features mentioned above are used to train a classifier via boosting a decision tree and for enforcing the space consistency over pixels; the condition random field is used in the training process. After the shadow is recognized, the proposed Gaussian model is used to remove the recognized shadow and the original shadow region is recovered.

E. Markov Random Field for Illumination Estimation

Panagopoulos *et al.* [21] proposed a novel framework to jointly recover the illumination condition from a single image, and this method needs the coarse geometry information of the given image. The described MRF illumination model estimates the illumination condition by combining the low-level shadow with high-level prior knowledge. Normally, the illumination could be estimated accurately by minimizing the MRF energy [42], [43]. However, this method is limited by the necessary prior knowledge.

III. APPROACH OVERVIEW

We first determine the Sun visibility of the given image by BLS, which could be expanded on the nodes in the width without the retraining process. Then, the useful information in the given image is divided into four categories in order to facilitate the follow-up prediction. The proposed information classification algorithm relies on a rough geometric structure and shadow area (see Section IV). These four categories are sky pixels, vertical surface, flake shadow, and long shadow. We use A to represent the sky pixel set. B represents the clique of the vertical objects with similar albedos and multiangle. C is a single object and its flake shadow. Long shadow is represented

---

Algorithm 1 Overview of Our Approach

Input: Input Images $I^1, I^2, \ldots, I^m$
Output: Final Shadow Synthetic Image
1: Determine the sun visibility by BLS;
2: Preliminary classify the useful information in every image by our *ICGS* algorithm;
3: Get the final classification A, B, C, D by Reclassify the information from all images;
4: Estimation illumination from each information class;
5: Eliminate the results of low possibility with small weight or dramatic difference from other estimating results;
6: Energy of our MRF model E for the synthesis;
7: **repeat**
8:     Calculate E of the possible estimating results;
9:     Record minimum energy E_{min} and associated result;
10: **until** All possible results calculated
11: Obtain the final synthesis result with minimum energy;

---

by D. We estimate the possible Sun position from the information clique A. The brightness of different orientation surfaces with similar albedo indicate the possible Sun light direction. The position relationship between the detected shadow and the original object may show the light direction. And the direction of the detected long shadow means the direction of the synthetic shadow. Finally, we integrate all of the estimating results from different cliques by MRF (see Section V).

As shown in Algorithm 1, m is the number of input images. And more images may mean more available information that can contribute to accurate estimation. We determine the Sun visibility at first. And, for the scene with visible Sun, all the estimating results from different kinds of information are compared to eliminate the estimating result with dramatic difference from other estimating results. The remaining estimating results will be integrated by MRF. We illustrate our approach in Fig. 2. The necessary input image (the top of the inputs) contains a lot of useful illumination cues. The common parts of the necessary input image and the auxiliary input images are considered as a characteristic point, which is used to calculate the rotation angle and displacement in the process of image stitching. First, the Sun visibility is determined by BLS, which could quickly make out the question and easily insert input nodes if necessary. Then, we preliminarily classify the information in each input image by the proposed ICGS for each image with visible Sun. The image pixels are classified into four classes: 1) sky pixels; 2) surface pixels (containing surfaces with shadows and without shadows); 3) long shadow with direction; and 4) cast shadow area with the original object. We will combine the information in every input image to get the final classification after rotating the angle of auxiliary input image(s). The estimating method of each information class is presented, and the illustration around the energy map is the information classified by ICGS. The final estimating result of our shadow synthesis is the possible shadow with the lowest energy.
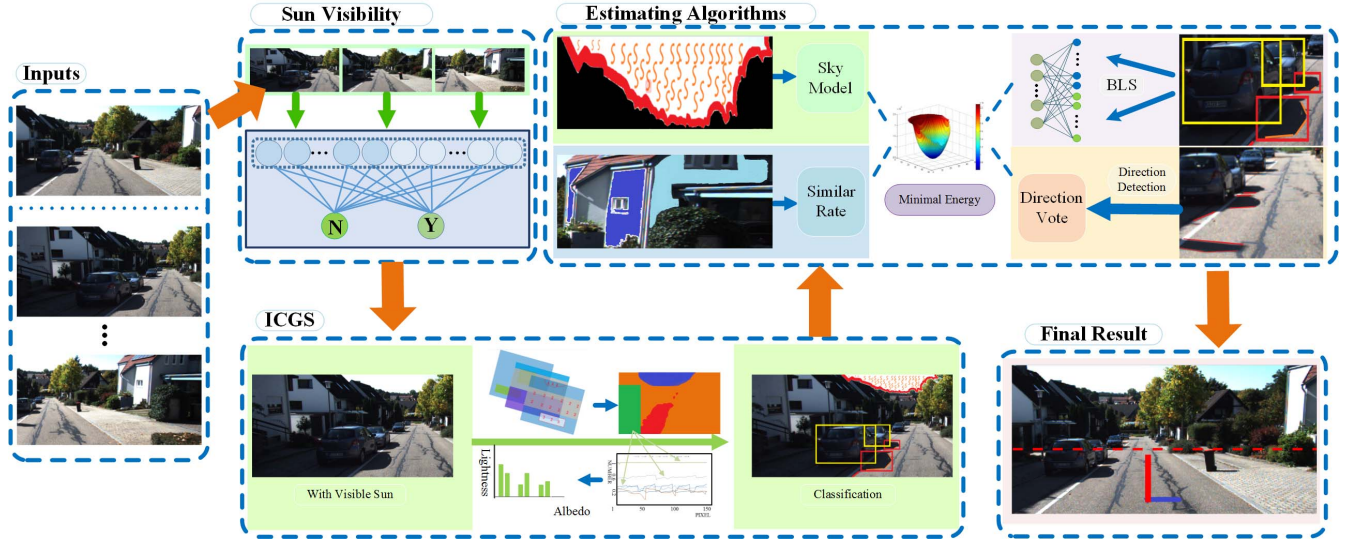
Fig. 2. This is the overview of the proposed shadow synthesis approach. The common parts of the input images are thought as characteristic point, which is necessary for calculating the rotation angle and displacement in the process of image stitching. The Sun visibility is determined by both basic and incremental BLS. ICGS is used to classify the corresponding information for each image with visible Sun. The estimating algorithms for each class information are used to estimate the illumination parameters. The final estimating result is obtained by MRF.
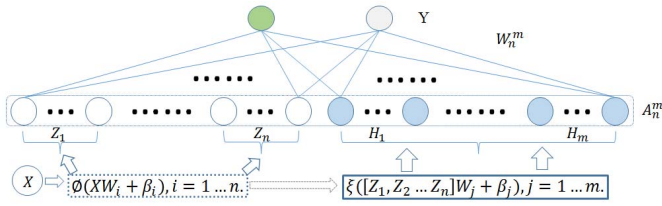

Fig. 3. Structure of BLS. The input layer is expressed as M_n^m, containing n groups of feature nodes Z and m groups of enhanced nodes H. The feature nodes Z_i are the mapped feature generated from the input data X, while the enhanced nodes H are the improved feature nodes.
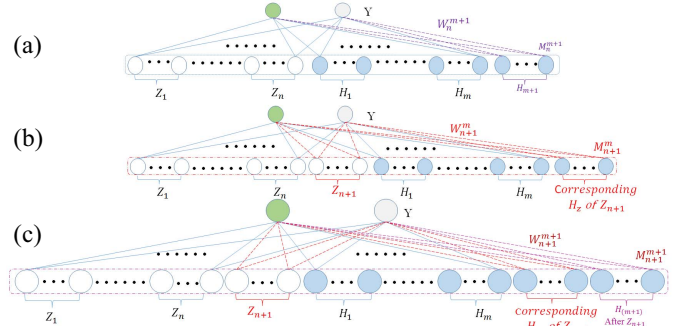

Fig. 4. Structure of incremental BLS. A group of additional enhanced nodes H_{m+1} are inserted in the original BLS, and connect the output nodes directly, as shown in (a). The increment of a group of feature nodes Z_{n+1} is illustrated in (b), and the corresponding enhanced nodes H_z are inserted here. (c) Structure of the increment of feature nodes and enhanced nodes synchronously, where the feature nodes are increased before enhanced nodes.

IV. SUN VISIBILITY AND INFORMATION CLASSIFICATION

As we assume that the Sun is the only light source in an outdoor scene, the Sun visibility has to be determined before the illumination estimating process. The appearance of the scene with visible Sun differs from the scene without visible Sun. And the result of the Sun visibility determining will directly affect the subsequent prediction of synthetic shadows. Different kinds of information about an image cannot be fully used to estimate the illumination parameters by the same algorithm, so we need to divide the image pixels into different categories. In order to reduce the error caused by the angle transformation, we extract the information from each image separately and then synthesize the clues through the angle transformation algorithm. We first classify the information in each image by our ICGS, and then we reclassify the information coming from different images for the final classification result.

A. Sun Visibility

1) Sun Visibility Classification Based on Basic BLS: The sky pixels and the shadow pixels may be indispensable for

determining the Sun visibility of an outdoor scene. The brighter sky pixels mean a higher possibility of visible Sun than the darker sky pixels. Obviously, the shadow pixels indicate a visible Sun as well. However, it is complex to get an accurate result of the Sun visibility quickly from those cues. In this paper, we determine the Sun visibility of the input images by basic BLS. As shown in Fig. 3, BLS is a flat network with only input nodes and output nodes. Its input nodes are composed of feature nodes and enhanced nodes. The feature nodes are computed from the input data X. There are two nodes in the final output layer Y, because we simply determine if there is a visible Sun in the input image by the basic image. We take a given outdoor image as the input data X of BLS, and the feature nodes Z_i are the mapping feature of the input data X with the transformation function ϕ. The ith mapped feature could be denoted as

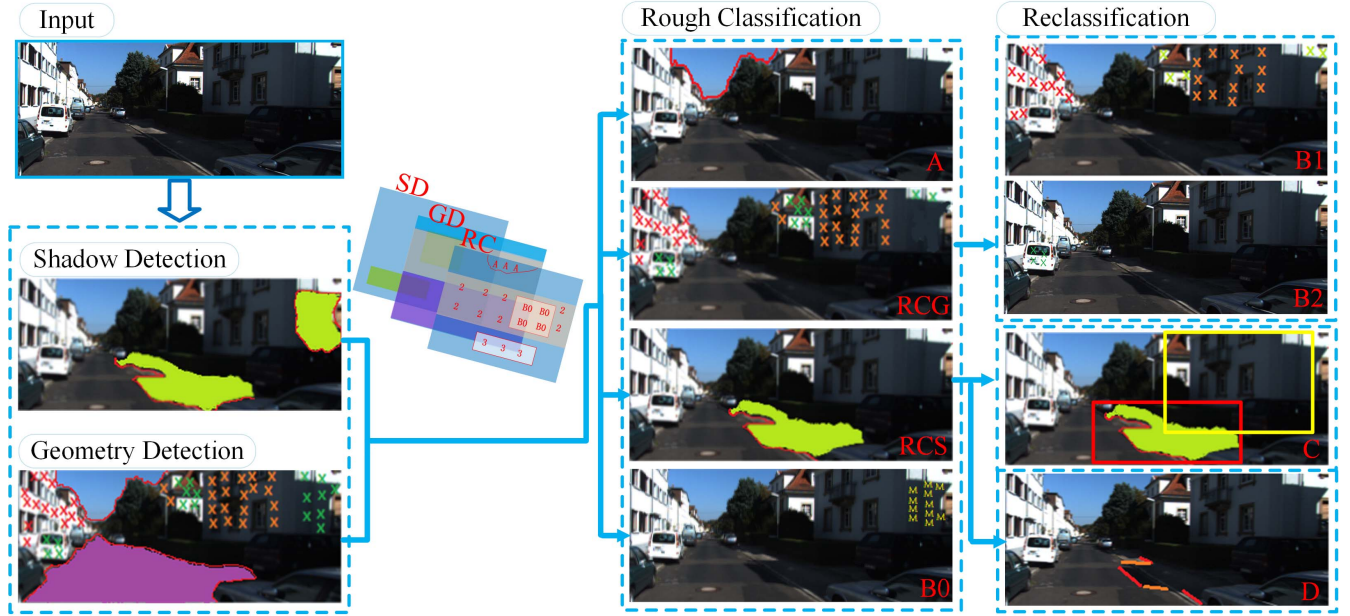$$Z_i = \phi_i(XW_i + \beta_i), \quad i = 1, \ldots, n \tag{1}$$

Fig. 5.   Algorithm of ICGS is shown here. For the input image, shadow area and the geometric structure are detected first. The geometric information (GD) is masked against the shadow information (SD), and RC illustrates the image segmentation based on the GD and SD. The rough classification is obtained by the intersection of shadow set and geometry set (ground and surface). A is the pixels in the sky area. RCG is the rough classification of useful surfaces. RCS is the rough classification of shadow on the ground. $B0$ is the shadow on the surface. The picture labeled by RCG is reclassified by albedo. $B1$ is the surfaces of house (with same albedo) with different angles, and $B2$ is the surface of the white car. The shadow on the ground is reclassified based on the shadow classification criterion. C is the flake shadow and its object, and D is the long shadow.

where ϕ_i is the transformation of ith mapped feature, W_i and β_i are generated randomly with proper dimensions. There are n groups of feature nodes, k nodes per group, and $n \times k$ feature nodes in total. For determining the Sun visibility from basic BLS, we set the number of feature nodes as 8×10. There are eight groups of mapping nodes and ten nodes per group.

To improve the learning ability of the flat network, the enhanced nodes are calculated by the mapping feature. The enhanced nodes H_j could be seen as the substitution for the hidden layers of the deep neural networks. The jth enhanced nodes could be denoted as

$$H_j = \xi_j([Z_1, \ldots, Z_n]W_j + \beta_j), \ j = 1, 2, \ldots, m \qquad (2)$$

where ξ_j is the transformation between jth enhanced node and all feature nodes Z_i. Similarly, W_j and β_j are generated randomly too. What have to be emphasized is that ϕ_i and ϕ_k could be different if $i \neq k$. Similarly, ξ_j and ξ_q could be different if $j \neq q$. There are m groups of enhanced nodes with q nodes per group, and we set $m = 1$ and $q = 7000$ in the Sun visibility determining process.

We represent all the input nodes with M_n^m, which contains n groups of feature nodes and m groups of enhanced nodes. The input nodes can be calculated by: $M_n^m = [Z_1, Z_2, \ldots, Z_n | H_1, H_2, \ldots, H_m]$. After the input nodes are established, there is the following formulation between the input nodes and output nodes: $Y = M_n^m W_n^m$. The weight coefficient W_n^m could be calculated by the pseudoinverse in BLS: $W_n^m = (M_n^m)^+ Y$, and the pseudoinverse of the input nodes $(M_n^m)^+$ could be calculated by: $(M_n^m)^+ = \lim_{\lambda \to 0} (\lambda I + M_n^m (M_n^m)^T)^{-1} (M_n^m)^T$. I is a unit matrix with proper dimensions and λ is a regularization parameter for ridge regression (see details in [31]) and λ is set as 10^{-8} here. Because the

weight coefficient could be calculated instead of the time-consuming iterative process, the system could be established quickly. We manually label 1300 images with $\widehat{V} = 1$ (with visible Sun) or $\widehat{V} = 0$ (without visible Sun) as the training samples. And then, we take 600 tapped images as the test samples. The training set and test set are split carefully to avoid the scene overlap. After the system is established, we test it by test samples. The test classification accuracy is 98.6% in our system. For the complex scene with overcast sky, the basic BLS cannot determine whether the Sun is visible based on the obvious shadow or cloudy sky. And the Sun visibility is supposed to be classified into more categories to express in detail. In this case, much more features are needed to classify the Sun visibility. The incremental BLS will be established to solve this complex condition.

2) Sun Visibility Classification Based on Incremental BLS: In the process of the classification by BLS, the nodes could be increased when the accuracy is not satisfied. While a time-consuming retraining process is followed by the incremental layers of deep learning structure, the new weight coefficient could be calculated by the original weight. Bluntly, based on the special architecture of BLS, the weights can be calculated without the retraining process. The enhanced node H, feature node Z, and the dimensions of the input data X can be increased individually or together in BLS. As shown in Fig. 4(a), a group of enhanced nodes H_{m+1} with q nodes is inserted. The additional nodes can be denoted as: $H_{m+1} = \xi_{m+1}([Z_1, \ldots, Z_n]W_{m+1} + \beta_{m+1})$, and then the updated input nodes M_n^{m+1} are denoted as

$$M_n^{m+1} = [Z_1, Z_2, \ldots, Z_n | H_1, H_2, \ldots, H_{m+1}]$$
$$\equiv [M_n^m | H_{m+1}]. \qquad (3)$$

Then, the pseudoinverse of the new input nodes $(M_n^{m+1})^+$ can be calculated by

$$\left(M_n^{m+1}\right)^+ = \begin{bmatrix} \left(M_n^m\right)^+ - \left(M_n^m\right)^+ H_{m+1} N^T \\ N^T \end{bmatrix} \qquad (4)$$

where N can be calculated by H_{m+1}, M_n^m (see details in [31]). Obviously, the weight W_n^{m+1} could be calculated by

$$W_n^{m+1} = \begin{bmatrix} W_n^m - \left(M_n^m\right)^+ H_{m+1} N^T Y \\ N^T Y \end{bmatrix}. \qquad (5)$$

The increment of feature nodes is necessary sometimes, because insufficient mapped feature may not extract enough underlying factors from the input data. As shown in Fig. 4(b), the corresponding enhanced nodes H_z increased after the additional $(n + 1)$th feature node group inserted. Similarly, the weights of the updated structure W_{n+1}^m can be calculated by W_n^m, Z_{n+1}, H_z, Y, and M_n^m. The increment of both the feature nodes and enhanced nodes is shown in Fig. 4(c). The additional enhanced nodes after the incremental feature nodes $H_{(m+1)}$ are different from H_{m+1}, where $H_{(m+1)} = \xi_{(m+1)}([Z_1, \ldots, Z_{n+1}]W_{(m+1)} + \beta_{(m+1)})$. Then, the new weights W_{n+1}^{m+1} can be calculated by twice the matrix conversion. While the increment molds mentioned above do not work well, the dimension of input data will be increased. The structure after the increment of X is similar to the original BLS structure, yet the dimensions of all matrices are changed. As before, the weights xW_n^m can be obtained by the original weight W_n^m.

For determining the Sun visibility of the input images, the classification result of basic BLS is accurate enough in our framework. The input image is classified into two classes: $\widehat{V} = 1$ or $\widehat{V} = 0$. Generally, the Sun visibility could not be simply classified into two different categories in nature images. The incremental BLS is used to classify the Sun visibility into more categories that can help to describe the illumination condition accurately. In follow-up work, we may need to judge the intensity of the illumination condition. The input images will be classified into 11 distinct categories: $\widehat{V} = 0$; $\widehat{V} = 0.1$; $\widehat{V} = 0.2$; $\widehat{V} = 0.3$; $\widehat{V} = 0.4$; $\widehat{V} = 0.5$; $\widehat{V} = 0.6$; $\widehat{V} = 0.7$; $\widehat{V} = 0.8$; $\widehat{V} = 0.9$; and $\widehat{V} = 1$. The incremental BLS may be necessary for the satisfied classification accuracy.

B. Information Classification Based on Geometry and Shadow

We proposed an information classification algorithm that is based on geometric structure and shadow area. ICGS divides the information into four categories. We detect the geometric structure by the rough geometry estimating algorithm [38], which can precisely divide the sky pixels, ground pixels, and geometry pixels from the image, and detect the orientation of the vertical surfaces. We first tap the image pixels with the label containing sky, ground, and vertical objects, represented as $L1$ after the geometric estimation. At the same time, we extract the shadow area using the method proposed in [40]. This method detects the shadow in a single outdoor image by checking the pixels around the edge. If the pixels satisfy the proposed three criteria, they are considered to be the shadow.

Algorithm 2 ICGS

Input: Image I, Lable1 $L1$, Lable2 $L2$
Output: Useful Information Cliques A, B, C, D
1: Detect the geometry and tab each pixel with Lable1 $L1$;
2: Detect the shadow and tab each pixel with Lable2 $L2$;
3: **for** Each image pixel p **do**
4:    Check $L1$ and $L2$
5:    $L1$ is Sky and $L2$ is non-shadow: $p_i \in A$;
6:    **if** $L1$ is ground and $L2$ is shadow **then**
7:      The shadow type satisfies Eq. (7): $p_i \in D$;
8:      There is a single object around the shadow: $p \in C$;
9:    **end if**
10:   **if** $L1$ is vertical object and $L2$ is shadow **then**
11:     $p \in B_0$;
12:   **end if**
13:   **if** $L1$ is vertical object and $L2$ is non-shadow **then**
14:     The pixel with albedo α_i, $i \in [1, 2, \cdots]$ belongs to the clique B_i;
15:     Mark B_i with three angles, as B_i^k, $k \in \{1, 2, 3\}$;
16:   **end if**
17: **end for**

The criteria have been derived by them from the new features they have just found. And we tap the picture with labels containing shadow and nonshadow, represented as $L2$ for each pixel. Then, we combine the two label results $L1$ and $L2$ to reclassify the information for making the best of the cues in the image. Algorithm 2 shows the details of algorithm ICGS. We divide the information into four classes by ICGS: 1) sky pixel clique (A); 2) clique of vertical surface with similar albedos or the surface with obvious shadow (B); 3) regions containing a single object with its shadow (C); and 4) long shadow clique (D). The process of ICGS is illustrated in Fig. 5. As we can see, the shadow area and geometric structure are detected from the input image first. Four different categories are classified based on the geometry and shadow, which is explained in detail in the following sections.

1) Categories of A and B: For estimating the illumination parameters, the distribution of the sky pixels in the image is very important. A different brightness distribution may indicate different position of the Sun. We extract the sky pixels and name the clique A in our approach. The clique A contains the brightness distribution of the sky pixels. The shadow detection of the outdoor scene image is extremely difficult because of the uncertain geometric structure, shadow-like objects, and changed weather conditions. Inaccurate shadow detection result will bring a confused estimation of the illumination, so the accurate shadow detection algorithm of the outdoor scene is necessary. We choose the shadow detection algorithm proposed in [40], which can accurately detect the shadow from a single image of the outdoor scene. Through the canny iterative calculations and checking the shadow features, the determined shadow region is more precise. This method ensures that the detected shadow is not the possible noise, such as the zebra crossing or the grass border area. After getting the shadow detection result, we will get the information class B. The shadow on the vertical object's surface may mislead the
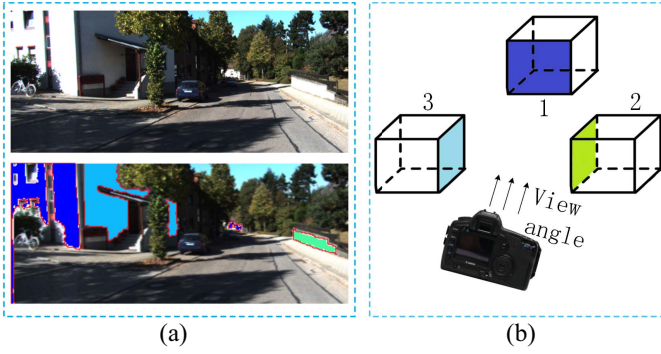
(a)　　　　　　　　(b)

Fig. 6.　In a nature scene, the possible angle of vertical objects can be divided into three classes, showing in (a). Thus, we assume that there are just three possible angles of vertical objects in the given image, which is shown in (b).

classification of different angle surfaces, and cause errors in the estimating result. By contraries, the surface with shadow means that it is impossible for the Sun to be in the opposite direction of the surface normal vector. Thus, we take the vertical surface with shadow as B_0. For the surface without shadow, we classify them based on different albedos. For surface with a particular albedo, we represent it as B_i, and $i \in [1, 2, \ldots]$.

For the surfaces with the same albedo in a different direction, the surface with higher brightness shows that the Sun light direction is parallel to its normal direction. Actually, the albedo of different surfaces in the same object is often similar. We use a popular object detection method [44] to determine which surfaces belong to the same object. For the surfaces of the houses, different surfaces of the same house may usually have the same albedo, and different houses in the same street may also have the same albedo. The surfaces of different houses in the given are divided into the same group B_i in this paper. Of course, before this step, the surfaces with shadow have already been divided into B_0. After selecting the brightness for each surface in B_i, we have to set an angle label for every surface and get a more detailed classification $B_i^k, k \in \{1, 2, 3\}$, which contains objects with similar albedo and particular orientation. In this paper, we assume that surfaces have just three kinds of orientations in the image and the others are needed to be classified into the nearest orientations. Fig. 6 shows how each angle is respective to the camera. We apply the numbers 1–3 to represent the angles 0°, 90°, and −90°. The three directions can be detected by the geometry detection algorithm.

Theoretically, the brightness of the surfaces with the same orientation and similar albedo should be the same (or similar), but there are some exceptions to the reality. The pixels of the glass on the window are different from the pixels of the wall surface; we call them special pixels, which should be eliminated. For the special pixels in B_i^k, we eliminate them by simply counting the number of different brightnesses, as shown in Fig. 7. The line of the same color represents the same surface, most of the pixel brightness within the same surface is in the vicinity of the same value, but a few are scattered in the vicinity of two to three values. We retain the brightness around the mode as the surface brightness and abandon the pixels with a significant difference from the representation (special pixels).
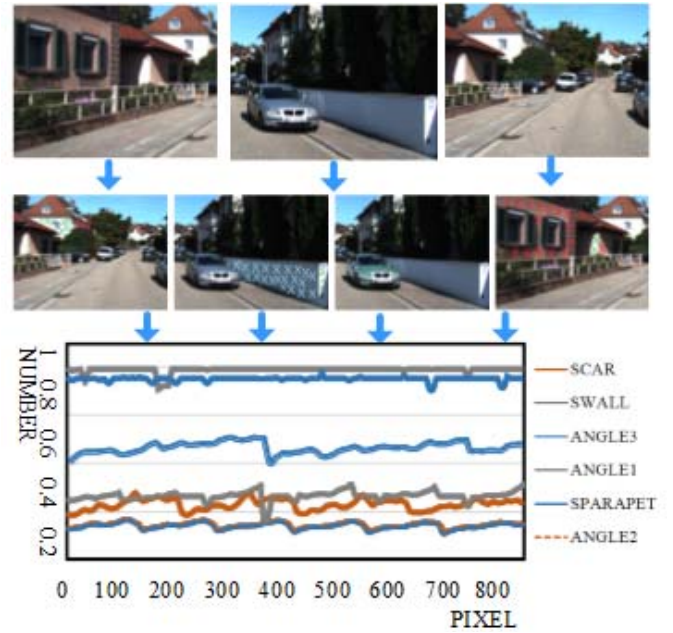

Fig. 7.　For the surface without shadow, we classify them according to the different albedo. The original images are shown at the top row, there are four kinds of surface in the input images. We take the pixels of house as "SWALL," and illustrate the value of pixels using gray lines. We take pixels of car as "SCAR," and illustrate the value of pixels using red lines. We take pixels of parapet as "SPARAPET," and illustrate the value of pixels using blue lines. The group number i of B_i is three here. We illustrate the pixels from different angle using different line types.

2) Categories of C and D: The difficulties in the process of making full use of the shadows include the shadow detection and the information extraction from the detected shadow. After detecting the shadow in the image by [40], we have to classify shadows on the ground to make the best of the shadows in the image. The shadow cast by a vertical object (such as a trunk, telegraph pole, etc.) infers a reasonable direction of the synthetic shadow. Thus, we want to pick this kind of shadow. From statistics of the world, long shadows are more likely to be cast by a vertical object. But the shadow cast by an object that is parallel to the ground (such as a roof or a wire) could be a long shadow too, and it may mislead the illumination estimation.

We first obtain a shadow on the ground by simply checking the pixel label. If the pixel's $L2$ is shadow and $L1$ is ground, the pixel will be defined as the shadow on the ground. We take the shadow pixels that are linked together as a shadow l_i according to the principle of continuity. Statistics of the world show that the longer shadow is more likely to be a desirous shadow than a shorter one. Thus, we confirm whether the shadow l_i belongs to the long shadow by

$$P(l_i \in D) = h_1 \cdot \left(\exp \frac{f(l_i)}{\sum_{p \in l_i} p} - 1 \right) + h_2 \cdot \mathcal{H}(g, l_i) \quad (6)$$

where h_1 is a positive weight coefficient (similar to $h_i, i \in [2, 9]$ below). For determining whether the shadow l_i could be divided into the long shadow clique D or not, we take the shadow shape feature, the strength of the shadow, and the geometry around the shadow into consideration in the formula.
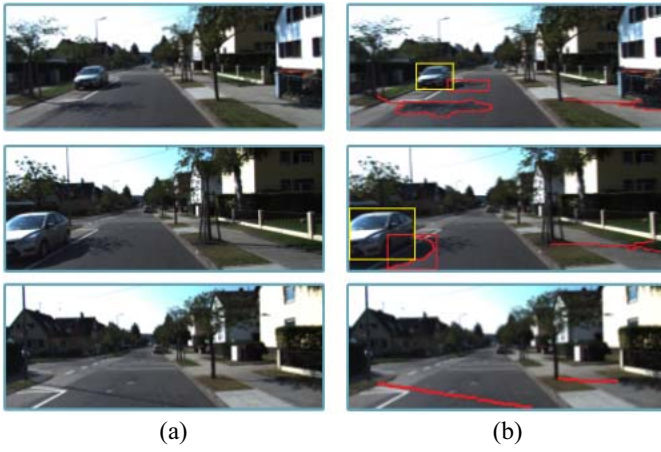
(a)   (b)

Fig. 8. Shadow on the ground in the input image is classified into two classes, represented as C and D. As shown in the right images, the set C contains cast shadow (red box) and the original object (yellow box), while set D contains long shadow (red line).

The function f counts the pixel numbers in one direction of the shadow l_i, and the results of the fraction between f and the pixel number of l_i may indicate if the shape of shadow l_i is a rectangle or square. As shown in (7), the function f represents the length of the shadow l_i, and it is calculated by counting the pixel in different directions and choosing the maximum number as the length of l_i. Spontaneously, the longest edge's direction β_i is taken as the direction of this shadow. If the shadow length is longer than the threshold $\tilde{l}$, we perform to take it into consideration. Otherwise, we would like to set $f(l_i)$ as 0. We take $\tilde{l}$ as 96 in our experiments

$$f(l_i) = \begin{cases} \max\left(\sum_{p\in l_i}^{\text{dir}1} p, \sum_{p\in l_i}^{\text{dir}2} p\right) & \text{if } f(l) \geq \tilde{l} \\ 0 & \text{if } f(l) \leq \tilde{l}. \end{cases} \quad (7)$$

In this situation, the main interference is the shadow cast by the object, which is parallel to the ground (such as a roof). If there is a vertical object around the shadow, the shadow is more likely to be in D. And for the function $\mathcal{H}(g, l)$, it may be defined as 0.5 if there is a vertical object around this shadow ($l_i \cap g \neq \varnothing$). Otherwise, it may be defined as 0 if $l_i \cap g = \varnothing$. And g is defined as the vertical objects set which can be obtained by the geometry detection algorithm. All of those conditions are considered measures of whether a shadow l_i can be classified into D. But even though a shadow $l_i \notin D$, it may be useful too. For example, a shadow of a motorcycle, to a great extent, does not belong to the long straight clique. But if the surrounding is not very complex, we can still use it by way of the BLS (see the details in Section V). The final shadow classification result is illustrated in Fig. 8 and the long shadow is classified into D while the flake shadow is classified into C. Finally, we divide the information into four classes by ICGS: A, B, C, and D, representing the sky pixel clique, clique of vertical surface (having multiangle) with similar albedos or the surface with obvious shadow, regions containing a single object with its shadow, and long shadow clique, respectively.
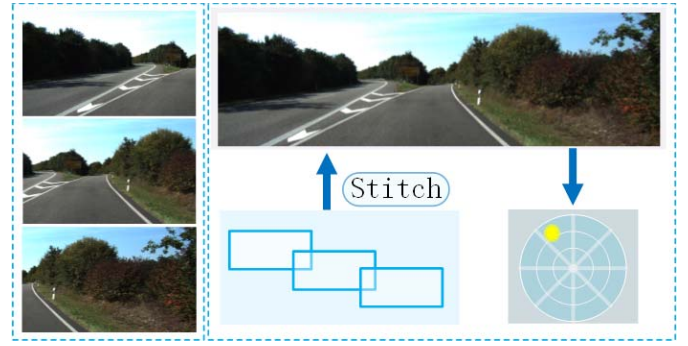

Fig. 9. We utilize the sky pixel set A to estimate the possible Sun position. First, we stitch the input images, we then extract all of sky pixel brightness and corresponding position as input of the sky model. Finally, we show the possible Sun position.

C. Coordinating Multiple Images Unity

Multiple images of the same scene may have many useful illumination cues. A single image may contain only one or two information classes of our four useful kinds. Under special circumstances, there may be no illumination cue in a single image. Even if a single image contains some illumination cues, the credibility of the estimating result may be reduced because of the incompleteness of the information. For the complex scene, the estimated result from a single image is not reliable because of the limited information. Thus, we take multiple images from different perspectives as the auxiliary inputs. While multiple images contain much more information for an accurate estimating result, the coordinate unification is the main problem of using multiple input image(s). In order to integrate the information of multiple images, we need to transform the coordinates of two pictures into the same coordinate system. The camera of the necessary input image is consistent with the world coordinate systems. In this paper, we use the method in [45] to achieve this goal.

Although the two images contain more information about illumination, the noises may be produced in the process if the coordinate of the two images are rotated first so we use the ICGS classification algorithm to process each input image first. We only obtain the corresponding pixel coordinates from the stitching result and, for the pixels in the gap between the images, we abandon them. After the coordinate unification, we reclassify all of the information classes. The sky pixels in the two images are more likely than the sky pixels in a single image. As shown in Fig. 9, we will take all of the sky pixels in the multi-images (after the coordinates are converted) as the final collection A. And for the cliques B, C, and D, we just simply convert the coordinate for the next prediction algorithms.

V. ESTIMATION ALGORITHMS AND OPTIMIZATION

A. Estimation Algorithm of A

According to the physical sky model [46], there is a certain correlation between the solar position and the sky pixels' brightness. And when the position of the Sun is known, the sky model gives the formula that can calculate the relative brightness of each position in the sky. The position of the Sun is represented by the Sun's zenith angle (θ_s) and the azimuth

angle (ϕ_s) with respect to the camera. In this paper, we represent the position of the Sun by using d_s instead of θ_s and ϕ_s. You can see d_s as a point in the sky, which could be constituted by all the nodes of a unit geodesic sphere. If we make discretization of zenith angle and azimuth angle, and we represent the number of ς as $N(\varsigma)$. The number of d_s is $N(d_s) = N(\theta_s) \cdot N(\phi_s)$. We utilize the difference between the brightness of the sky pixels at different positions, which can be known from the image, to determine the position of the Sun. First, we have to check the clique A, the sky pixels set in the image to determine whether the physical model is suitable or not. If the sky is clouded or overexposed, J is set to 0. And it means that the sky pixels are abandoned. Otherwise, $J = 1$. The position of the Sun can be calculated by

$$\max_{(d_s)} J \exp\left(\sum_{a_i \in A}(a_i - h_3 \cdot \mathcal{G}(d_s, \Omega_c))^2\right) \quad (8)$$

where a_i is the brightness of the sky pixel in a particular position, and the function $\mathcal{G}$ is the sky model function for solving the relative brightness of the sky pixel. The maximum likelihood solar position is obtained by a satisfied d_s. Ω_c is the parameter set, containing the coordinates of pixel and camera, which can be calculated from the image. Obviously, a particular solar position corresponds to a specific light direction, and we will directly use the light direction $d_{\hat{l}}$ to indicate the illumination in the following section.

Every sky pixel is checked whether it conforms to the *Perez* model [46] or not, because there may be overexposure and sky pixels rarely not suitable for judging the illumination condition. If the sky pixels do not comply with the position of the Sun, we give up the clique A.

B. Estimation Algorithm of B

According to the *ICGS* classification algorithm, the clique B contains vertical surfaces with different angles and similar albedo. In order to simplify the algorithm, we divide the direction of the surface by its normal vector into three categories, and the others are divided into the nearest orientation. $B_i^k, k \in [1, 3]$ contain the brightness of surface pixels with a particular angle. For the surfaces B_i with only one orientation ($k = 1$), we will abandon them because there is no comparison to show which orientation is much brighter. We define the similarity between the direction of the surface's normal (γ_i^j) and the direction of the light source ($d_{\hat{l}}$)

$$\mathcal{S}\left(\gamma_i^k, d_{\hat{l}}\right) = \frac{1}{1 + e^{\left(-h_4 \cdot \sum_{b_j \in B_i^j} b_j + h_5\right)}} \quad (9)$$

where b_j is the brightness of a particular pixel and h_3 and h_4 are no less than 0. Brighter pixels in B_i^j mean that its normal direction is similar to the direction of the light source. Thus, we compare $\mathcal{S}(\gamma_i^1, d_{\hat{l}})$ with $\mathcal{S}(\gamma_i^2, d_{\hat{l}})$ and $\mathcal{S}(\gamma_i^3, d_{\hat{l}})$, and choose the smallest one as the basis to calculate the rate of the other two to basis. For instance, if the $\mathcal{S}(\gamma_i^1, d_{\hat{l}})$ is the smallest one, the rate of $\mathcal{S}(i)_1^2 = \mathcal{S}(\gamma_i^2, d_{\hat{l}})/\mathcal{S}(\gamma_i^1, d_{\hat{l}})$ and $\mathcal{S}(i)_1^3 = \mathcal{S}(\gamma_i^3, d_{\hat{l}})/\mathcal{S}(\gamma_i^1, d_{\hat{l}})$ is calculated. And for all of i, we calculate the summary of $\mathcal{S}(i)_1^2$ and $\mathcal{S}(i)_1^2$ to integrate the
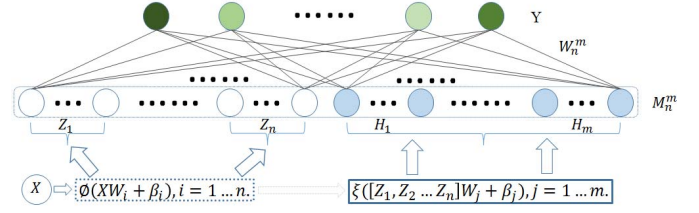

Fig. 10. Structure of BLS. The input layer is expressed as M_n^m, containing n groups of feature nodes Z_i and m groups of enhanced nodes H_j. The feature nodes Z_i are the mapped feature generated from the input data X, while the enhanced nodes H are the improved feature nodes. There are eight nodes in the final output layer Y, because we are supposed to divide the input images into eight categories, and each type means the possible shadow direction within a range of 45°.
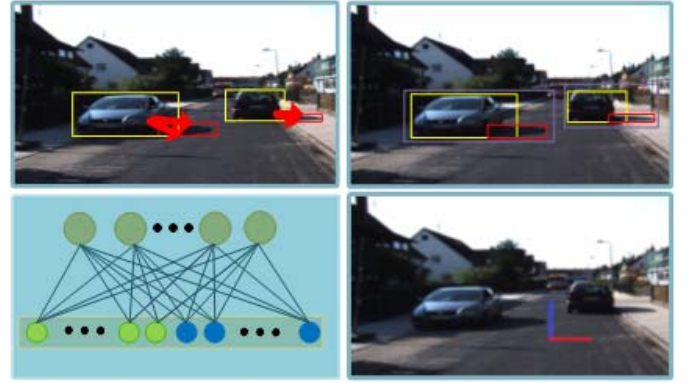

Fig. 11. Estimating result of information class C is shown in the lower-right corner. The detected object and its cast shadow is shown in the middle of the input images (upper-left). The red arrows show the orientation from object to shadow.

surface with different albedos. Finally, $\sum_i \mathcal{S}(i)_1^2$ is compared with $\sum_i \mathcal{S}(i)_1^2$ to determine the illumination direction.

C. Estimation Algorithm of C

After the classification algorithm *ICGS*, we get a kind of information containing an object and its shadow. From the object position corresponding to the cast shadow area, the possible light direction could be estimated. We apply the BLS to estimate the possible angle of the synthetic shadow from clique C. As shown in Fig. 10, the structure is similar to Fig. 3 but the number of input nodes M_n^m and output nodes Y are increased. Because of the uncertain shadow shape and the complex geometric structure of the clique C, it is difficult to estimate an accurate shadow direction. If the object is on the left of the cast shadow area, the light direction may be on the left side, left front, or left rear. We divide the possible synthetic shadow direction 360° (from $-180°$ to $180°$) into eight copies and each part contains a range of 45°. As shown in Fig. 11, the indicated direction (red arrowhead) from the position of the object and its cast shadow is shown at the upper-left. We classify the image into one of the eight different classes by the BLS.

The image pixels in the purple box at the upper right are taken as the input data X. We set the number of feature nodes Z_n as 100 and set the number of enhanced nodes H_m as 11 000. We manually label 3600 images with $D_{\hat{l}}$ as the training samples, and the training samples are shown in Fig. 12. $D_{\hat{l}}$
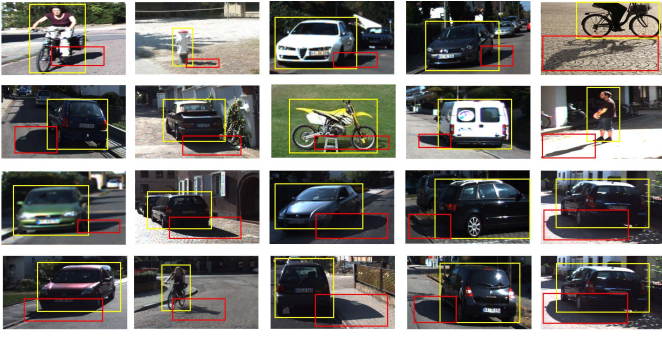
Fig. 12.   Training samples.

represents the most likely light direction in the input images. We take 800 labeled images as test samples. $D_{\hat{l}} \in 1, 2, \ldots, 8$ means a different angle of the synthetic shadow

$$d_{\hat{l}} \in \left[\frac{(Y-1) \cdot \pi}{4}, \frac{(Y-1) \cdot \pi}{4} \right] \qquad (10)$$

where $d_{\hat{l}}$ is the direction of light and Y is the output with maximum possibility. The objects in the training images contain pedestrians, cars, bicycles, motorcycles, and so on. The accuracy is up to 96.7%. Of course, this estimation algorithm will be useless if there is no class C in all input images. Finally, the input image is classified into one of the eight classes with particular orientation.

D. Estimation Algorithm of D

We judge whether a shadow could be used to determine the synthesis' shadow direction of the virtual object by (6). For the $l_i \in D$, we pretend to take the direction β_i of l_i or $\beta_i + \pi$ as the light direction

$$\mathcal{F}(\beta_i, d_{\hat{l}}) = \min \left(\frac{(\beta_i - d_{\hat{l}})^2}{\pi}, \frac{((\beta_i + \pi) - d_{\hat{l}})^2}{\pi} \right). \qquad (11)$$

In this equation, we determine the synthetic shadow direction by determining the minimal angle difference to the possible light direction, which could be determined by (10).

It is precise because this method can directly obtain the direction of the shadow. If the false direction β is exiting in the input data, the result is fatal. Before determining the possible synthetic shadow direction, we eliminate the possible noise by voting of all the shadow l_i. If the shadow is cast by a clothesline pole (or a rope across the air), it may cause a detected false direction. But the voting process will eliminate the error caused by this shadow because most of the shadows l_i in D are cast by vertical objects. This method may be misleading. For example, the shadow of an object might be casted on a wall, and the wall is actually built in the vertical direction which could be detected. This condition will lead to the direction of the shadow rotation. Therefore, the direction of the shadow in this paper still holds that there is a certain possibility of the shadow direction.

E. Information Results Integrated by MRF

After getting rough estimating results from four different weak clues, we will combine them by MRF. We will assume

the surfaces in the scene exhibit Lambertian reflectance, which is a commonly used assumption. The outgoing radiance at a particular pixel p is given by

$$\mathcal{I}(p) = \alpha_p \big(V_0 + V(p, d_{\hat{l}}) \max\{d_{\hat{l}} \cdot n_p, 0\} \big) + \omega \qquad (12)$$

where $\mathcal{I}(p)$ is the brightness of the pixel p and α_p is the albedo of this pixel. If the ray from the point p along the light direction $d_{\hat{l}}$ intersects the geometric structure, $V(p, d_{\hat{l}})$ will be defined as 0. Otherwise, $V(p, d_{\hat{l}})$ will be defined as 1. V_0 is the ambient intensity of the scene and ω is the possible noise. Although the point p is not occlusive by the geometry structure, it still could not be lighted. If the multiplication of light direction $d_{\hat{l}}$ and the normal of point n_p is less than 0, p cannot be lighted. For the pixels in the shadow area, the brightness $\mathcal{I}_s(p)$ can be expressed by the ambient intensity V_0. And $\mathcal{I}_s(p)$ is the rough approximation of the synthetic shadow. Because the pixel $\mathcal{I}_s(p)$ could be obtained in the input image, the brightness could be roughly determined by V_0.

In this paper, the final estimating result is obtained by sampling in a large probability range. The probability range could be obtained based on the estimating result from the four kinds of information. We minimize the energy of the synthesized object for the final estimating result. The energy of our model consisting of pixel nodes and light nodes is defined as

$$E(\hat{x}) = \sum_{i \in S_{\hat{l}}} \phi_s(\hat{x}_i) + \phi_{\hat{l}}(\hat{x}_{\hat{l}}) + \sum_{(i,j) \in \Psi} \psi_\Psi(\hat{x}_i, \hat{x}_j) + \sum_{i \in P} \psi_p(\hat{x}_i, \hat{x}_{\hat{l}})$$

$$(13)$$

where $\phi_s(\hat{x}_i)$ and $\phi_{\hat{l}}(\hat{x}_{\hat{l}})$ are the singleton potentials for pixel nodes and light nodes, $\psi_\Psi(\hat{x}_i, \hat{x}_j)$ and $\psi_p(\hat{x}_i, \hat{x}_{\hat{l}})$ are the pairwise potential defined on two pixels and associating light node with a pixel $\hat{x}_i$, respectively. The pixel set $S_{\hat{l}}$ represents the pixels contained in the synthetic shadow, P is the pixel set of all the pixels in the image, and $\hat{x}_i$ is the brightness of pixel i.

The singleton potential $\phi_p(\hat{x}_i)$ is the difference between the synthetic pixel and the value of shadow pixel in given image

$$\phi_s(\hat{x}_i) = h_6 \cdot \min \big(|\hat{x}_i - \mathcal{I}_s(i)|, \tilde{t}_p \big) \qquad (14)$$

where the upper bound $\tilde{t}_p$ is the cost term for avoiding over-penalizing outliers. The singleton potential $\phi_\Psi(\hat{x}_{\hat{l}})$ is the difference between the gradient angle of synthetic shadow edge $\tau_s(i)$ and the gradient angle of original shadow edge $\tau_r(i)$

$$\phi_\Psi(\hat{x}_{\hat{l}}) = h_7 \cdot \frac{1}{|\Psi_{S_{\hat{l}}}(\hat{x}_{\hat{l}})|} \sum_{i \in D} \frac{h_8 \cdot |\tau_r(i) - \tau_s(i)|}{2\pi} \qquad (15)$$

where $\Psi_{S_{\hat{l}}}(\hat{x}_{\hat{l}})$ is the edge set of the shadow cast by the light node $\hat{x}_{\hat{l}}$. Let $\tau_s(i) \in [0, 2\pi)$ be similar to $\tau_r(i) \in [0, 2\pi)$ to favor the similarity of the direction of the synthetic shadow and the original shadow. The relation between pixel node $\hat{x}_i$ and light node $\hat{x}_{\hat{l}}$ is defined as

$$\psi_p(\hat{x}_i, \hat{x}_{\hat{l}}) = h_9 \big(\alpha_{\hat{x}_i}(V_0 + V(\hat{x}_i, d_{\hat{l}}) \max\{d_{\hat{l}} \cdot n_{\hat{x}_i}, 0\}) - \hat{x}_i \big)^2 + \tilde{t}_c$$

$$(16)$$

where an upper bound $\tilde{t}_c$ is set to avoid overfitting and $d_{\hat{l}}$ is the light direction of the light node $\hat{x}_{\hat{l}}$. We minimize the energy to determine the final result from the possible light direction $d_{\hat{l}}$,
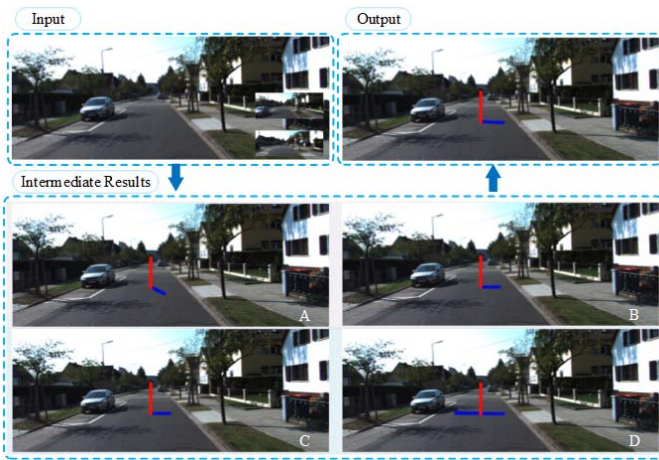
Fig. 13. For the input image, the estimating result of each class information is shown here, and the result by integrating four classes of information is illustrated at last.
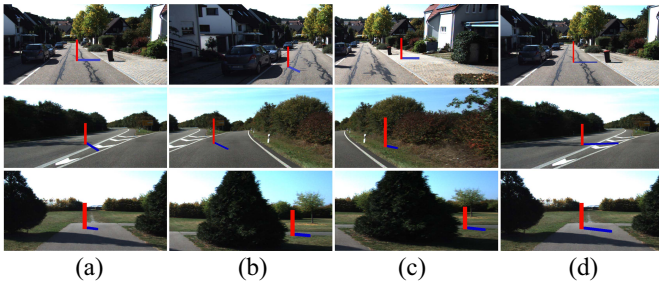

Fig. 14. Estimating result of single image is shown in (a)–(c), and the final result of all input images (the cues in necessary input image (a) and the auxiliary images (b) and (c) are integrated) is shown in (d).

which could be obtained by different information classes of the input images. The final estimating result depends on the similarity of the pixel brightness and the original image.

For illustrating the result of integration, we show the result obtained by each information class and the integration of those results by minimizing the energy. As shown in Fig. 13, the result of each information class is not reliable. The sky pixels in the input image are not enough for the prediction of an accurate Sun position. There are surfaces with two different orientations of houses in the image, so the estimated direction of B is the normal of the brighter surface. The long shadows indicate two possible directions. The final result is more accurate than the results obtained by each information class.

VI. EXPERIMENTAL RESULTS

We evaluate the effectiveness of our framework in two different ways. First, we show our experimental results and compare with the previous methods, which are done in similar conditions. All of the experiments are tested on a MATLAB software platform on a laptop equipped with Intel-i5 2.4GHz CPU, 16GB memory. The information contained in a single image containing outdoor scenes is not accurate in most cases. At that time, we predicted the illumination conditions. We use auxiliary input images (including the same parts of the
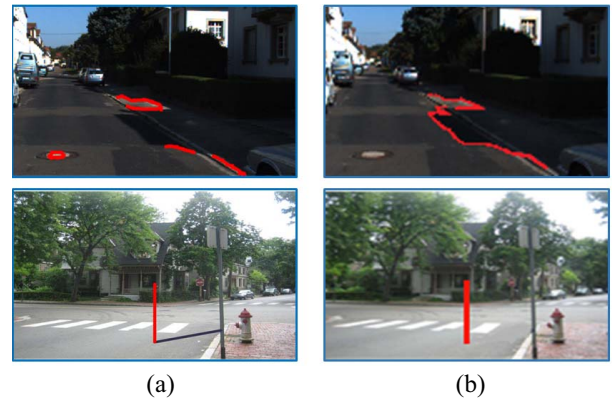

Fig. 15. Shadow detection result comparison between our approach and Lalonde *et al.* [23]. The shadow detection and the synthetic shadow of Lalonde *et al.* [23] are shown in (a), and the result of ours is shown in (b).
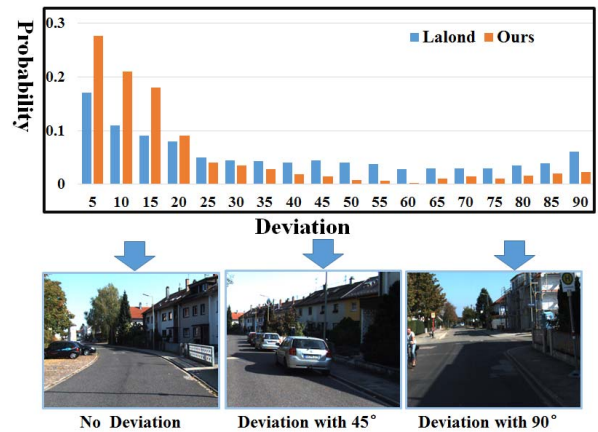

Fig. 16. Orientation deviation of the long shadow by our approach and Lalonde *et al.* [23] method.

necessary input images and the same scenes) to improve the prediction results, as shown in Fig. 14.

A. Comparison With State-of-the-Art

While the exiting shadow plays an important role in the illumination estimating process, the accuracy of the shadow detection and the shadow exploitation directly impacts the prediction result. As shown in Fig. 15, Lalonde *et al.* [23] may mistake the objects on the road mistake as shades, and the estimated result of the scene without visible Sun is based on the angle of the zebra crossing as in Fig. 15(a). And because of the inaccurate shadow detection, the illumination estimating result is wrong. Since we chose the long shadow based on the shadow shape and the geometry structure, the long shadow is cast by vertical trees to a great extent and the illumination estimating result is more accurate. The deviation of the estimated results based on the orientation of long shadow is shown in Fig. 16; the proportion estimated with low deviation is higher than Lalonde *et al.* [23]. As shown in Fig. 17, our results (second row) is more accurate than others (top row). Fig. 17(a) is the result comparison with Lalonde *et al.* [23]. Fig. 17(b) is the result comparison with Lalonde *et al.* [37]. Fig. 17(c) is the result comparison with Panagopoulos *et al.* [21].

CHEN *et al.*: OUTDOOR SHADOW ESTIMATING USING MULTICLASS GEOMETRIC DECOMPOSITION BASED ON BLS 2163

TABLE I
PARAMETERS CALCULATED BY [47]. SE IS THE ELEVATION OF SUN, SA
IS THE AZIMUTH OF SUN, CE IS THE CAMERA ELEVATION, AND
VF IS THE VERTICAL FIELD OF VIEW

Image	SE (radians)	SA (radians)	CE (radians)	VF (degrees)
Image1	1.100	1.473	0.258	48.516
Image2	0.471	-0.295	0.002	48.980
Image3	0.471	1.080	-0.014	42.019

TABLE II
SUN VISIBILITY COMPARISON

Methods	Accuracy (%)		Training Time (s)	
	SV	DC	SV	DC
SAE	98.35	96.42	37446.25	38634.31
DBN	98.71	96.82	56231.72	62351.61
FRBM	97.46	96.38	632.43	684.32
BL	**98.64**	**96.73**	**42.65**	**48.71**
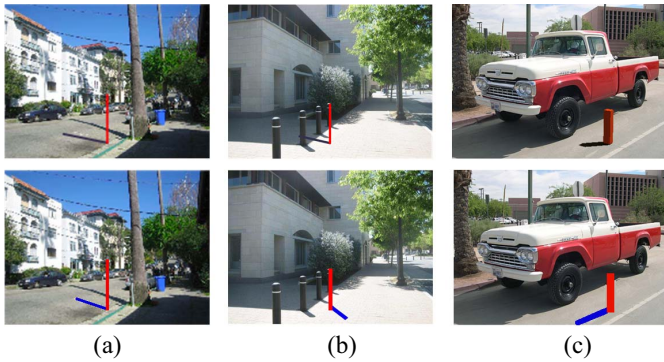
Fig. 17. Estimated result comparison with [21], [23], and [37]. Top row consists of the estimated result of [23] (left), [37] (middle), and [21] (right). The second row is the result of ours from single input image.
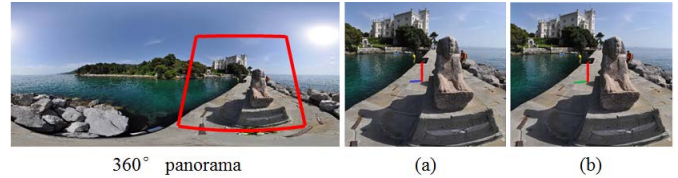

Fig. 18. Comparison with [47] on 360° panorama (left) from the SUN360 panorama dataset [48]. Our estimating result is shown in (a) and the result of [47] is shown in (b).
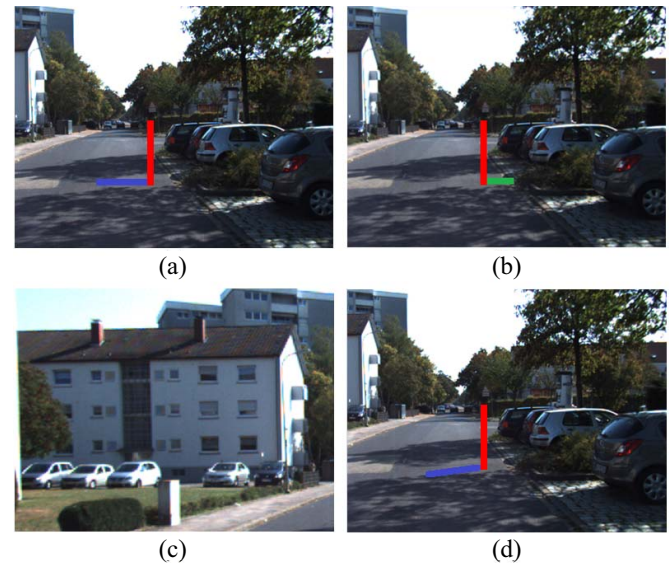

Fig. 19. Comparison between our approach and [47] on nature image. The illumination estimation result of our approach from a single image is shown in (a). (b) is drawn based on the parameters from [47]. (d) Estimating result of our approach, which integrates the information of necessary input image (a) and the auxiliary input image (c).

Recently, Hold-Geoffroy *et al.* [47] proposed a learning-based illumination estimation method. They train the CNN with large set of 360° panoramas from the SUN360 panorama dataset [48]. And we compare our estimation result with [47] on both image in 360° panorama (Fig. 18) and nature image (Fig. 19). As shown in Fig. 18, the estimating result of our approach is shown in Fig. 18(a), and the estimation result of [47] is shown in Fig. 18(b). The estimating result of [47] is drawn by the parameters (the elevation of the Sun, the azimuth of the Sun, the camera elevation, and vertical field of view) calculated by [47]. The values of these parameters are shown in Table I, and "Image1" represents Fig. 18(b), "Image2" represents Fig. 19(b), and "Image3" represents Fig. 19(c); the parameters of "Image3" have not been illustrated in the image. We can see clearly from Fig. 18 that our estimated result tends to differ very little from the estimated result of [47] on the image from the SUN360 panorama dataset [48].

For the nature image with complex geometry, [47] erroneously estimated the illumination condition totally [Fig. 19(b)]. The azimuth of the Sun is −0.295 (radians). The Sun is on the left of the image while the Sun should be on the right in real life. Compared with [47], our results based on a single image [Fig. 19(a)] is closer to the real illumination condition. If we could get Fig. 19(c) as an auxiliary input image, the illumination estimation result of our method [Fig. 19(d)] is more reasonable. As we can see in Table I, the azimuth of the Sun estimated from Fig. 19(c) is similar to the real illumination condition but the accurate estimating result could not help to refine the estimating result [Fig. 19(b)]. The accuracy and training time comparison with others are shown in Table II, "SV" means Sun visibility and "DC" means the direction classification. As shown in Table II, the accuracy of

BLS is similar to other deep learning systems but the training time is much quicker than others.

VII. CONCLUSION

We have demonstrated a complete framework for the illumination estimation using the corresponding information existing in a given outdoor image. Our approach determines the Sun visibility by BLS, which can be expanded quickly to determine the Sun visibility of a complex scene. Then, we divide the useful information by the proposed classification method, which is based on the geometric structure and the shadow area. The information classification algorithm can help make full use of the information in the next steps. After the classification, we estimate the illumination from every category of information

by respective algorithms. Finally, for an effective estimating result, we combine the intermediate estimating results obtained by MRF. Experimental results show that our approach predicts the illumination conditions accurately on a wide range of images. We compare our results with other methods in similar conditions and the comparison results show that our approach has a more effective prediction ability. Furthermore, we compare the learning-based illumination estimating method, which shows our estimating result is better than the learning-based method by the auxiliary input image(s). As for future work, we plan to utilize the relationship between pixels and the illumination condition in the image, such as the shadow area without certain directions and adjacent vertical objects, so that our method can solve a wider range of images.

REFERENCES

[1] M. J. Landau, B. Y. Choo, and P. A. Beling, "Simulating Kinect infrared and depth images," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3018–3031, Dec. 2016.

[2] S. Wang, B. Pan, H. Chen, and Q. Ji, "Thermal augmented expression recognition," *IEEE Trans. Cybern.*, vol. 48, no. 7, pp. 2203–2214, Jul. 2018.

[3] P. Jiang, Y. Cheng, X. Wang, and Z. Feng, "Unfalsified visual servoing for simultaneous object recognition and pose tracking," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3032–3046, Dec. 2016.

[4] X. Huang, X. Wang, J. Gao, and R. Yang, "Estimating pose and illumination direction for frontal face synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2008, pp. 1–6.

[5] S. Luo, H. Li, and H. Shen, "Shadow removal based on clustering correction of illumination field for urban aerial remote sensing images," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 485–489.

[6] P. Li, H. Sun, C. Huang, J. Shen, and Y. Nie, "Interactive image/video retexturing using GPU parallelism," *Comput. Graph.*, vol. 36, no. 8, pp. 1048–1059, 2012.

[7] P. Li, H. Sun, B. Sheng, and J. Shen, "Image stylization with enhanced structure on GPU," *Sci. China Inf. Sci.*, vol. 55, no. 5, pp. 1093–1105, 2012.

[8] T. Luo, C. Hou, F. Nie, and D. Yi, "Dimension reduction for non-Gaussian data by adaptive discriminative analysis," *IEEE Trans. Cybern.*, to be published.

[9] J. Shen, X. Yang, X. Li, and Y. Jia, "Intrinsic image decomposition using optimization and user scribbles," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 425–436, Apr. 2013.

[10] A. Subpa-Asa, Y. Zheng, N. Ono, and I. Sato, "Light transport component decomposition using multi-frequency illumination," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 3595–3599.

[11] K. Takechi and T. Okabe, "Diffuse-specular separation of multi-view images under varying illumination," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 2632–2636.

[12] J. Yu and Z.-F. Wang, "A video, text, and speech-driven realistic 3-D virtual head for human–machine interface," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 991–1002, May 2015.

[13] Y. Nie, H. Sun, P. Li, C. Xiao, and K. Ma, "Object movements synopsis via part assembling and stitching," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 9, pp. 1303–1315, Sep. 2014.

[14] W. Zhang, G. Li, and Z. Ying, "A new underwater image enhancing method via color correction and illumination adjustment," in *Proc. IEEE Vis. Commun. Image Process.*, 2017, pp. 1–4.

[15] X. Zhang and X. Wu, "Illumination invariant feature based on neighboring radiance ratio," in *Proc. IEEE Vis. Commun. Image Process.*, 2017, pp. 1–4.

[16] N. McLaughlin, J. Ming, and D. Crookes, "Largest matching areas for illumination and occlusion robust face recognition," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 796–808, Mar. 2017.

[17] X. Zhao, G. Evangelopoulos, D. Chu, S. Shah, and I. A. Kakadiaris, "Minimizing illumination differences for 3D to 2D face recognition using lighting maps," *IEEE Trans. Cybern.*, vol. 44, no. 5, pp. 725–736, May 2014.

[18] B. Sheng *et al.*, "Retinal vessel segmentation using minimum spanning superpixel tree detector," *IEEE Trans. Cybern.*, to be published.

[19] C. Jenila and R. K. Jeyachitra, "Performance analysis of indoor joint illumination and communication systems using light emitting diodes and laser diodes," in *Proc. Int. Conf. Wireless Commun. Signal Process. Netw.*, 2017, pp. 905–910.

[20] F. Xie, L. Tao, and G. Xu, "Estimating illumination parameters using spherical harmonics coefficients in frequency space," *Tsinghua Sci. Technol.*, vol. 12, no. 1, pp. 44–50, Feb. 2007.

[21] A. Panagopoulos, C. Wang, D. Samaras, and N. Paragios, "Illumination estimation and cast shadow detection through a higher-order graphical model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 673–680.

[22] B. Sheng, P. Li, Y. Jin, P. Tan, and T. Lee, "Intrinsic image decomposition with step and drift shading separation," *IEEE Trans. Vis. Comput. Graphics*, to be published.

[23] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Estimating the natural illumination conditions from a single outdoor image," *Int. J. Comput. Vis.*, vol. 98, no. 2, pp. 123–145, 2012.

[24] M. Sun, Z. Zhou, Q. Hu, Z. Wang, and J. Jiang, "SG-FCN: A motion and memory-based deep learning model for video saliency detection," *IEEE Trans. Cybern.*, to be published.

[25] A. Kamel *et al.*, "Deep convolutional neural networks for human action recognition using depth maps and postures," *IEEE Trans. Syst., Man, Cybern., Syst*, to be published.

[26] A. Karambakhsh *et al.*, "Deep gesture interaction for augmented anatomy learning," *Int. J. Inf. Manag.*, to be published.

[27] Y. Wen, B. Sheng, P. Li, W. Lin, and D. D. Feng, "Deep color guided coarse-to-fine convolutional network cascade for depth image super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 994–1006, Feb. 2019.

[28] A. Masood *et al.*, "Computer-assisted decision support system in pulmonary cancer detection and stage classification on CT images," *J. Biomed. Informat.*, vol. 79, pp. 117–128, Mar. 2018.

[29] F. Jiang *et al.*, "Abdominal adipose tissues extraction using multi-scale deep neural network," *Neurocomputing*, vol. 229, pp. 23–33, Mar. 2017.

[30] B. Sheng, P. Li, C. Gao, and K. Ma, "Deep neural representation guided face sketch synthesis," *IEEE Trans. Vis. Comput. Graphics*, to be published.

[31] C. L. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.

[32] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 125–138, Jan. 2016.

[33] W. Hou, X. Gao, D. Tao, and X. Li, "Blind image quality assessment via deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1275–1286, Jun. 2015.

[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[35] C. L. P. Chen and Z. Liu, "Broad learning system: A new learning paradigm and system without going deep," in *Proc. Youth Acad. Annu. Conf. Chin. Assoc. Autom.*, 2017, pp. 1271–1276.

[36] Z. Liu, J. Zhou, and C. L. P. Chen, "Broad learning system: Feature extraction based on K-means clustering algorithm," in *Proc. Int. Conf. Inf. Cybern. Comput. Soc. Syst.*, 2017, pp. 683–687.

[37] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan, "Estimating natural illumination from a single outdoor image," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2009, pp. 183–190.

[38] D. Hoiem, A. A. Efros, and M. Hebert, "Geometric context from a single image," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 1, 2005, pp. 654–661.

[39] C. Wang, L. Deng, Z. Zhou, M. Yang, and B. Wang, "Shadow detection and removal for illumination consistency on the road," in *Proc. Int. Conf. Security Pattern Anal. Cybern.*, 2017, pp. 198–203.

[40] J. Tian, X. Qi, L. Qu, and Y. Tang, "New spectrum ratio properties and features for shadow detection," *Pattern Recognit.*, vol. 51, pp. 85–96, Mar. 2016.

[41] M. Xu *et al.*, "Learning-based shadow recognition and removal from monochromatic natural images," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5811–5824, Dec. 2017.

[42] Y. Tao, Y. Shen, B. Sheng, P. Li, and R. W. H. Lau, "Video decolorization using visual proximity coherence optimization," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1406–1419, May 2018.

[43] Y. Nie, C. Xiao, H. Sun, and P. Li, "Compact video synopsis via global spatiotemporal optimization," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 10, pp. 1664–1676, Oct. 2013.

[44] P. Ren, W. Fang, and S. Djahel, "A novel YOLO-based real-time people counting approach," in *Proc. Int. Smart Cities Conf.*, 2017, pp. 1–2.
[45] Z. Wei, S. Dai, and C. Lin, "Research on adaptive fusion algorithm for image stitching," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst.*, 2017, pp. 270–274.
[46] R. Perez, R. Seals, and J. Michalsky, "All-weather model for sky luminance distribution—Preliminary configuration and validation," *Solar Energy*, vol. 50, no. 3, pp. 235–245, 1993.
[47] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde, "Deep outdoor illumination estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2373–2382.
[48] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba, "Recognizing scene viewpoint using panoramic place representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2695–2702.
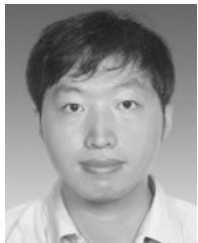
Ping Li received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong.

He is currently an Assistant Professor with the Macau University of Science and Technology, Macau, China. His current research interests include image/video stylization, GPU acceleration, and creative media. He has one image/video processing national invention patent, and has excellent research project reported worldwide by *ACM TechNews*.

Zhihua Chen received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China.

He is currently a Full Professor with the Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai. His current research interests include image/video processing and computer vision.
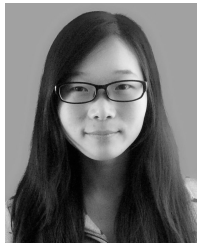
Ting Gao received the B.Eng. degree in computer science from the East China University of Science and Technology, Shanghai, China, where she is currently pursuing the M.Eng. degree in computer science with the Department of Computer Science and Engineering.

Her current research interests include illumination estimating, Markov random field, shadow synthesis, broad learning system, and computer vision.

C. L. Philip Chen (S'88–M'88–SM'94–F'07) received the M.S. degree in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 1985, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1988.

He is currently a Chair Professor with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau, China. Being a Program Evaluator of the accreditation board of engineering and technology education in the U.S., for computer engineering, electrical engineering, and software engineering programs, he successfully architects the University of Macau's Engineering and Computer Science programs receiving accreditations from Washington/Seoul Accord through Hong Kong Institution of Engineers (HKIE), of which is considered as his utmost contribution in engineering/computer science education for Macau as the former Dean of the Faculty. His current research interests include systems, cybernetics, and computational intelligence.

Dr. Chen was a recipient of the 2016 Outstanding Electrical and Computer Engineers Award from his alma mater, Purdue University, after he graduated from the University of Michigan at Ann Arbor, Ann Arbor, MI, USA in 1985. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS and an Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS and the IEEE TRANSACTIONS ON CYBERNETICS. He was the Chair of TC 9.1 Economic and Business Systems of International Federation of Automatic Control from 2015 to 2017. He was the IEEE SMC Society President from 2012 to 2013 and is currently a Vice President of the Chinese Association of Automation (CAA). He is a member of Academia Europaea and the International Academy of Systems and Cybernetics Science and a fellow of AAAS, IAPR, CAA, and HKIE.

Bin Sheng received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong.

He is currently an Associate Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include machine learning, virtual reality, and computer graphics.