**ORIGINAL PAPER**

# Car model recognition by utilizing symmetric property to overcome severe pose variation

**Hui-Zhen Gu · Suh-Yin Lee**

**Abstract** This paper presents a mirror morphing scheme to deal with the challenging pose variation problem in car model recognition. Conventionally, researchers adopt pose estimation techniques to overcome the pose problem, whereas it is difficult to obtain very accurate pose estimation. Moreover, slight deviation in pose estimation degrades the recognition performance dramatically. The mirror morphing technique utilizes the symmetric property of cars to normalize car images of any orientation into a typical view. Therefore, the pose error and center bias can be eliminated and satisfactory recognition performance can be obtained. To support mirror morphing, active shape model (ASM) is used to acquire car shape information. An effective pose and center estimation approach is also proposed to provide a good initialization for ASM. In experiments, our proposed car model recognition system can achieve very high recognition rate (>95%) with very low probability of false alarm even when it is dealing with the severe pose problem in the cases of cars with similar shape and color.

**Keywords** Car model recognition · Pose estimation · Active shape model · Mirror morphing · Template matching

## 1 Introduction

Personal automobiles are popular vehicles, therefore a recognition system capable of identifying car models

H.-Z. Gu (✉) · S.-Y. Lee
Department of Computer Science and Information Engineering,
National Chiao Tung University, 1001 Ta Hsueh Rd.,
Hsinchu 300, Taiwan
e-mail: hcku@cs.nctu.edu.tw

S.-Y. Lee
e-mail: sylee@csie.nctu.edu.tw

automatically would be useful for many applications such as object tracking and traffic surveillance. Unfortunately, it is not easy to identify car models, because vehicles in the same class tend to appear similar. In addition, the fact that cars would be viewed from different orientations also compounds the difficulty of car model recognition.

In conventional vehicle recognition systems, license plate numbers are recognized [1–3], so vehicles can be identified and then car models can be acquired. However, license plates may be changed or hidden, particularly in criminal cases. Typical appearance-based vehicle recognition approaches utilize appearances such as size [4], shape [5–7], or color [8,9] to identify different cars. Car model can be identified by their unique features, such as the ratio of width to height, segmented binary shape, and the binary edges map.

These appearance-based methods usually suffer from the pose variation problem which will deform the shape of the appearances dramatically. Pose problem is a thorny issue in most object recognition fields, including face recognition [10–13]. Many current approaches [6,7] of car recognition are applicable only from a particular viewpoint. In [6], a camera is installed in front of a parking lot entrance, so that every image would be captured in the front view or rear view. In [7], a camera is mounted in a car driving down a freeway, so most of the cars are captured from behind. Such limitations on specific views make the car model recognition systems impossible to deal with many general cases, for instance, cars maneuvering or driving through an open field.

To enable car recognition without viewpoint constraints, some researchers [14–19] estimate the pose first and then match the image to a specific template with the same pose as the estimated one. One drawback of these approaches is that slight deviation in pose estimation degrades the recognition performance dramatically. Nonetheless, it is difficult to estimate the pose of a car with a high degree of accuracy

[16–19]. In [20], a quasi-rigid alignment combined with a flexible matching method is proposed to deal with pose variation for car identification. This method identifies cars correctly if they have distinctly different shapes or colors. However, it still has the difficulty in discriminating a car from other cars with similar shape and color. Therefore, finding an effective algorithm for car model recognition to accommodate the severe pose variation problem in the cases of similar shape and color would be of great value and interest.

This study utilizes the symmetric property of cars to overcome this problem. In previous studies, symmetry was used to track, locate, or recognize vehicles [21–23] by identifying the central line of tested cars. Subsequent researches [24–26] improved the means by which symmetry is measured. Since these methods rely heavily on "explicit" symmetric properties found in the images of cars, they are generally applicable only within the narrow orientation of front or rear view. When a car is oriented in a wider view, the explicit symmetry may be attenuated in the car image. Therefore, in severe pose situations, such symmetry-based methods are barely able to differentiate one car from other cars with a similar shape and color.

To improve recognition performance in severe pose variation cases, this paper proposes a mirror morphing scheme. The scheme produces the mirror image of the non-symmetric image of a tested car and then synthesizes the car image and its mirror image to form a symmetric representation. Based on knowledge concerning the inherent symmetry of the tested car (not the image of the tested car), the correspondence relationship between the shape features found in the tested car image and those of its mirror image can be established. Then through the process of morphing, the synthesized image appears in an exactly typical (front, rear or side) view, with the symmetric axis positioned precisely in the middle of the image. When the synthesized image of the car is matched against the template image for car model recognition, the factors influencing recognition accuracy, such as pose error and center bias are eliminated. Therefore, satisfactory recognition performance in situations of widely divergent poses can be obtained.

Mirror morphing requires knowing the locations of shape feature of cars first. Active shape model (ASM) [27,28] is adopted to extract the shape features of cars to support mirror morphing. To obtain a good initialization for ASM, an effective algorithm to estimate vehicle pose and acquire center position initially is also proposed. The estimated initial pose and center prepares ASM a good start toward the right direction to appearances location. The techniques: pose and center estimation, shape feature extraction by ASM, and mirror morphing are integrated to form up an advanced system for car model recognition.

The remaining sections are organized as follows. Section 2 provides the system overview. Sections 3, 4, 5 describe the primary functions of our proposed system: pose and center estimation, shape feature extraction, and mirror morphing, respectively. Section 6 describes the experiments and Sect. 7 gives the conclusions.

## 2 System overview

The flowchart of the proposed car model recognition system is depicted as in Fig. 1. It comprises five main steps: (1) pose and center estimation, (2) shape feature extraction, (3) car region segmentation, (4) mirror morphing, and (5) template matching. These five steps are described briefly in this section and described in greater details in Sects. 3, 4 and 5, respectively.

The pose and center estimation step calculates the pose angle ($\theta$) and center position ($pos_{cen}$) of a car. A car image can be extracted from video frames by frame difference [29, 30] or background subtraction [31,32] as a bounding rectangle. In the rectangle, the symmetric center in the front (or rear) region of the car is extracted. The ratio that divides the distance between the symmetric center and the left (or right) rectangle end by the rectangle width is calculated. An approach which estimates the pose angle by the ratio and utilizes the symmetric center as the reference position for matching is proposed.

The shape feature extraction step searches the appearance features of cars despite the geometric deformation when vehicles orientate. Key points located at the corners or boundaries of the windshield, headlight, or bumper on the car image, are extracted to comprise the shape features. To extract the shape features automatically and efficiently, ASM [27,28] is adopted. ASM generates a statistical model using the key points labeled on training images. Then the trained model is iteratively deformed to find the corresponding key points on a tested image. ASM is suitable to find the shape features of objects with obvious boundaries, such as cars. Compared with some advanced modeling techniques [33,34], the 2D model technique, such as ASM, is faster and more practical.

Since the information outside the typical (front, rear, or side) view region is useless and usually causes interference for mirror morphing, the car region segmentation step performs a separating process prior to mirror morphing to exclude those uninteresting regions. By the shape features acquired by ASM, the car region surrounding the outer contours of the shape features can be separated.

The mirror morphing step provides the core function of our proposed system. If the view of a tested car is limited in one of the typical (front, rear and side) view, correct recognition of car models would be easier to achieve. Hence, the mirror morphing technique, which transforms the segmented image from any pose into a typical view, is

**Fig. 1** System flowchart for car model recognition
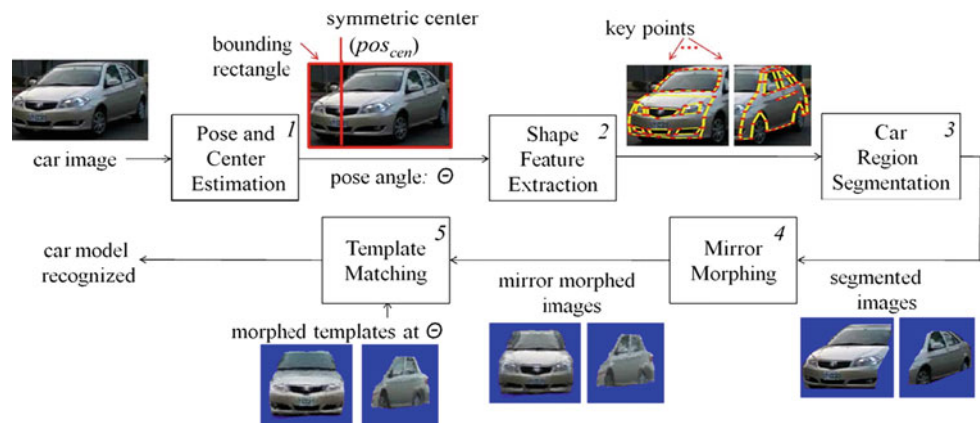


proposed. The mirror image of the segmented image is produced through mirror transformation. The symmetric correspondence relationship between the segmented image and its mirror image can be derived from the result of the shape feature extraction step. Then, these two mutually symmetric images are synthesized into a typical view by the general morphing technique [35,36]. Since the pose variation and center bias can be eliminated through mirror morphing, the tested car can be recognized more accurately in the next step.

In the template matching step, the template car image with orientation closest to the estimated pose angle ($\theta$) is selected and it is also processed by mirror morphing. The synthesized tested car image is matched against the morphed template car image pixel by pixel. The matching error between the two images in RGB color space is calculated. If the matching error is less than a predefined threshold, the tested car is deemed to be the same model as the template car. Otherwise, the tested car and the template car are considered different car models.

Since the mirror morphing process is more applicable on images in the same view, the car image is separated into two regions: front and side regions (or rear and side regions). These two regions are processed simultaneously through the five functional steps in Fig. 1. First, the pose angle and center position of the front (or rear) region are estimated, and those of the side region are derived from the estimated results of the front (or rear) region. Second, the shape features on the two regions (front + side or rear + side) are extracted by ASM separately. Third, a car image is segmented into a front view image and a side view image (or a rear view image and a side view image). Next, the mirror morphing individually synthesizes the front (or rear) and the side images. Finally, each synthesized image is matched against the template of the same view and at the same pose.

When the pose angle of a car is close to 0° (or 180°), the front (or rear) region provides dominant information required for car model recognition. If the pose angle is close to 90°
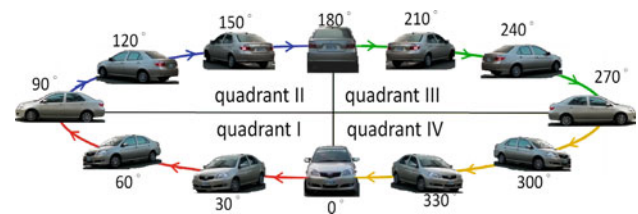


**Fig. 2** Four quadrants referring to vehicle orientations

(or 270°), the information of the side region will be dominant. In between these four typical orientations, both of the front and side regions (or the rear and side regions) provide contributions to recognition. Because the side region is not horizontally symmetric as the front (or rear) region is, the proposed techniques of symmetric center extraction and mirror morphing cannot be directly applied. The modifications are individually described in Sects. 3 and 5.

## 3 Pose and center estimation

This section addresses the pose and center estimation step in greater details. Generally, the left, right, top, and bottom ends ($pos_{left}$, $pos_{right}$, $pos_{top}$, $pos_{bot}$) of the bounding box of a car can be extracted by frame difference [29,30] or background subtraction [31,32]. The geometric center generated by the four ends is deemed as the reference position for matching. The ratio of the car width (wid $= pos_{right} - pos_{left}$) to car height (hgt $= pos_{bot} - pos_{top}$) denoted by $\gamma_1$ is obtained as Eq. (1):

$$\gamma_1 = \frac{\text{wid}}{\text{hgt}} = \frac{pos_{right} - pos_{left}}{pos_{bot} - pos_{top}}. \tag{1}$$

In Fig. 2, all orientations (pose angles) can be divided into four quadrants: (I: 0°−90°, II: 90°−180°, III: 180°−270°, and IV: 270°−360°). For each quadrant, we collected a set of standard templates with known pose angles ($\theta$) and manually

**Fig. 3  a** The four ends, the symmetric center, height, width and half width of a car image, **b** the planar area and the possible range to search for the symmetric center, **c** the check width and the check point at $x$-axis = 27, **d** the check point at 43 and the pixels (*red* to *red*, *blue* to *blue*, *green* to *green*) in correspondence to compute ASD, **e** ASD results and the prominent factor results for all check points (color figure online)
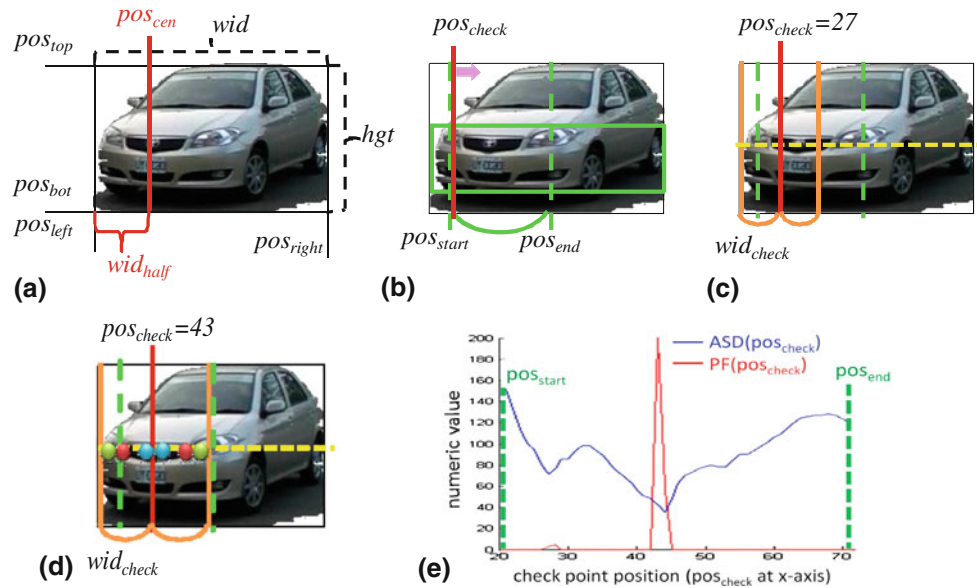


extracted the ratios on these templates. A monotonic function Ratio$_1$ (where $\gamma_1 = \text{Ratio}_1(\theta)$) can be constructed by these ratios for each quadrant. To estimate the pose angle $\theta_1 (\theta = \theta_1)$ of the tested car in a quadrant, the width and height of the car are extracted and the ratio $\gamma_1$ is computed. Then, the pose angle of the tested car can be computed using the inverse function of Ratio$_1$ as Eq. (2):

$$\theta_1 = \text{Ratio}_1^{-1}(\gamma_1). \tag{2}$$

Because various car models have different car lengths and widths, the Ratio$_1$ function belonging to a specific target car model can be used to estimate the pose angle in a quadrant. To extend the processing into multi-model recognition, as when identifying $N$ target car models, $N$ Ratio$_1$ functions are trained in each quadrant. An estimated pose angle is produced by each Ratio$_1$ function and $N$ estimated pose angles are obtained in this pose and center estimation step. After the processes of the subsequent steps of the proposed system, the model of the tested car is finally decided in the template matching step.

For each target car model, if no prior information concerning the quadrant is available, the computed ratio $\gamma_1$ of the tested car image corresponds to four possible pose angles in the four quadrants. To determine which quadrant is the correct one, four template images of the target car model at the four pose angles are chosen. The tested car image is then matched against the four template images and the template image with the minimum matching error is selected in this pose and center estimation step. The quadrant and pose angle of the selected template image are considered those of the tested car.

The approach to estimate car pose utilizing $\gamma_1$ is termed as the conventional pose estimation approach. In this approach,

the bottom ends are easily interfered by some environmental effects, such as shadows. The ratio of the width to height may also be erroneous if a small tilt rotation between the camera and the vehicle exists. Hence, this approach generally cannot provide accurate enough pose angles.

We searched for further information to improve pose estimation accuracy. The symmetric center pos$_{cen}$ in the front (or rear) region of a car shown in Fig. 3a could be a good choice. The symmetric center can be viewed as a reference point for matching the car, and the half width wid$_{half}$ of the front (or rear) region can be utilized to estimate the pose angle. When the car is orientated within quadrant I or III, the symmetric center is searched beginning from left toward right and the half width can be defined as wid$_{half}$ = pos$_{cen}$ − pos$_{left}$. Similarly, if the car is orientated within quadrant II or IV, the algorithm in searching the symmetric center should begin from right toward left and the half width is defined as wid$_{half}$ = pos$_{right}$ − pos$_{cen}$.

A new ratio of wid$_{half}$ to wid, denoted by $\gamma_2$, is proposed in Eq. (3) for estimating the pose angle of the tested car. For each target car model, a set of standard templates with known pose angles ($\theta$) was collected and the ratios $\gamma_2$ of the templates were manually extracted. In each quadrant, a monotonic function Ratio$_2$ (where $\gamma_2 = \text{Ratio}_2(\theta)$) can be constructed by the template ratios. Each target car model obtains its unique function Ratio$_2$ in a quadrant. To estimate the pose angle $\theta_2 (\theta = \theta_2)$ of the tested car, the ratio $\gamma_2$ is first obtained, whereupon the pose angle of the tested car can be computed by the inverse function of Ratio$_2$ as Eq. (4). In multi-model recognition, as when identifying $N$ target car models, $N$ Ratio$_2$ functions are required to be trained in each quadrant. Similar to the processing of Ratio$_1$, $N$ estimated pose angles are obtained in the pose and center estimation step

step. The model of the tested car is finally decided in the template matching step.

$$\gamma_2 = \frac{\text{wid}_{\text{half}}}{\text{wid}}$$

$$= \begin{cases} \frac{\text{pos}_{\text{cen}} - \text{pos}_{\text{left}}}{\text{pos}_{\text{right}} - \text{pos}_{\text{left}}}, & \text{if } \theta \in [0°, 90°] \text{ or } [180°, 270°] \\ \frac{\text{pos}_{\text{right}} - \text{pos}_{\text{cen}}}{\text{pos}_{\text{right}} - \text{pos}_{\text{left}}}, & \text{if } \theta \in [90°, 180°] \text{ or } [270°, 360°] \end{cases} \quad (3)$$

$$\theta_2 = \text{Ratio}_2^{-1}(\gamma_2). \quad (4)$$

In the proposed pose estimation approach adopting $\gamma_2$, if no prior information about the quadrant is available, a symmetric center candidate on the left side and another candidate on the right side are sought simultaneously. Two half widths ($\text{wid}_{\text{half}} = \text{pos}_{\text{cen}} - \text{pos}_{\text{left}}$ or $\text{pos}_{\text{right}} - \text{pos}_{\text{cen}}$) and two ratios $\gamma_2$ are computed. Each ratio corresponds to two pose angles in two quadrants: (I and III), or (II and IV), so two ratios produce four possible pose angles in the four quadrants for each target car model. The template images belonging to each target car model depicting the four possible poses by $\gamma_2$ are acquired. The tested car image is matched against the four template images and the template image with the minimum matching error is selected in the pose and center estimation step. The quadrant and pose angle of the selected template image are adopted as those of the tested car.

To estimate the pose angle, the symmetric center has to be obtained. Traditional symmetry-based methods [21–26] were developed to acquire the symmetric centers of cars within a narrow range of orientations. To accommodate wider range of poses, a more robust algorithm to identify the symmetric center is proposed. Figure 3b–e shows an example to introduce the algorithm.

First, we select an area on about the same plane as the green rectangle in Fig. 3b and check the symmetric property of the pixels in this area. Since the pose angle can be coarsely estimated by the four ends of the car, a possible range [$\text{pos}_{\text{start}}, \text{pos}_{\text{end}}$] which the symmetric center may appear is assigned. In Fig. 3c, for each row in the selected area, a check point ($\text{pos}_{\text{check}}$) moves gradually from $\text{pos}_{\text{start}}$ to $\text{pos}_{\text{end}}$ and the check width ($\text{wid}_{\text{check}} = \text{pos}_{\text{check}} - \text{pos}_{\text{left}}$ or $\text{pos}_{\text{right}} - \text{pos}_{\text{check}}$) increases. For each check point, the accumulated symmetric difference (ASD) is computed as follows.

$$\text{ASD}\left(\text{pos}_{\text{check}}\right)$$

$$= \begin{cases} \sum_{x=\text{pos}_{\text{left}}}^{x=\text{pos}_{\text{check}}} \frac{|\text{Int}(2 \times \text{pos}_{\text{check}} - x, y) - \text{Int}(x, y)|}{2 \times (\text{pos}_{\text{check}} - \text{pos}_{\text{left}})}, \\ \quad \text{if } \theta \in [0°, 90°] \text{ or } [180°, 270°] \\ \\ \sum_{x=\text{pos}_{\text{right}}}^{x=\text{pos}_{\text{check}}} \frac{|\text{Int}(2 \times \text{pos}_{\text{check}} - x, y) - \text{Int}(x, y)|}{2 \times \left(\text{pos}_{\text{right}} - \text{pos}_{\text{check}}\right)}, \\ \quad \text{if } \theta \in [90°, 180°] \text{ or } [270°, 360°] \end{cases} , \quad (5)$$

where $\text{Int}(x, y)$ is the intensity value at the pixel $(x, y)$ on a tested car image, and $\text{pos}_{\text{check}} \in [\text{pos}_{\text{start}}, \text{pos}_{\text{end}}]$.

In Fig. 3d, the red pixel in the left side of $\text{pos}_{\text{check}}$ corresponds to the red pixel in the right side. In the same way, the same color pixels in the left side and the right side of $\text{pos}_{\text{check}}$ are in correspondence. The intensity differences of the corresponding pixels are accumulated to acquire the ASD. Figure 3e shows the ASD results over all check points as the blue curve. Because the ASD of the symmetric center should be relatively lower than those of neighboring pixels, a valley would appear in the ASD distribution. The valley test is defined in Eqs. (6) and (7).

$$\text{VT}_1(\text{pos}_{\text{check}}) = \sum_{i=1}^{k_{\text{vt}}} \text{ASD}(\text{pos}_{\text{check}} - i) - k_{\text{vt}} \times \text{ASD}(\text{pos}_{\text{check}}) \quad (6)$$

$$\text{VT}_2(\text{pos}_{\text{check}}) = \sum_{i=1}^{k_{\text{vt}}} \text{ASD}(\text{pos}_{\text{check}} + i) - k_{\text{vt}} \times \text{ASD}(\text{pos}_{\text{check}}), \quad (7)$$

where $k_{\text{vt}}$ is the predefined range of the neighboring pixels for the valley test.

Only the check points passing the valley test (i.e., $\text{VT}_1(\text{pos}_{\text{check}}) > \text{th}_{\text{vt}}$ and $\text{VT}_2(\text{pos}_{\text{check}}) > \text{th}_{\text{vt}}$, where $\text{th}_{\text{vt}}$ is a predefined threshold) are considered candidates for the symmetric center. In Fig. 3e, it is obvious that ASD values decrease gradually toward the valley, and then increase gradually when leaving the valley. Therefore, the idea of calculating $\text{VT}_1$ and $\text{VT}_2$ is the same as calculating the triangular area in the range of $k_{\text{vt}}$, if the $x$-axis is shifted up to the ASD value of $\text{pos}_{\text{check}}$. The valley test threshold $\text{th}_{\text{vt}}$ is designed by $\lfloor (k_{\text{vt}} \times k_{\text{vt}})/2 \rfloor$, which represents the area of an isosceles right triangle with side length $k_{\text{vt}}$. Empirically, the value of $k_{\text{vt}}$ is 5 and the threshold $\text{th}_{\text{vt}}$ will be 12. Based on the results of the experimental trials, the threshold can provide satisfactory results. In Fig. 3e, the check points at $x$-axis 27 and 43 pass the valley test, so they are candidates of the symmetric center.

Due to local uniform intensity values, the check point ($\text{pos}_{\text{check}} = 27$) is a valley; however, it is not the actual symmetric center. To reject this false report, a prominent factor test is defined as Eq. (8).

$$\text{PF}(\text{pos}_{\text{check}}) = 2 \times k_{\text{pf}} \times \text{Int}(\text{pos}_{\text{check}})$$

$$- \sum_{i=1}^{k_{\text{pf}}} \text{Int}(\text{pos}_{\text{check}} - i)$$

$$- \sum_{i=1}^{k_{\text{pf}}} \text{Int}(\text{pos}_{\text{check}} + i), \quad (8)$$

where $k_{pf}$ is the predefined range for the prominent factor test.

A number of distinguishable objects, such as logos and license plates, commonly appear along the central line in the front region of cars. These objects obviously differ in intensity from the neighboring pixels and can be utilized to determine whether the check points occur in the region of uniform intensity. The prominent factor sums the differences in intensity between $2 \times k_{pf}$ neighboring pixels and the check point. If the check point has a sufficiently large prominent factor (i.e., $|PF(pos_{check})| > th_{pf}$, where $th_{pf}$ is a predefined threshold), this check point is considered a candidate symmetric center with high priority. The prominent factor test works only for the check points that pass the valley test. If multiple valley points pass the prominent factor test, the valley point with the minimum ASD is selected as the symmetric center of the current row. In the case that no valley point passes the prominent factor test, the valley point with the minimum ASD is still selected as the symmetric center of the current row.

Because the tested car is identified with each specific target car model, the prominent factor threshold of a target car model can be defined specifically. In an extreme case, a small threshold can be set if the target car model does not have distinguishable objects along the central line. In our experience, human eyes are capable of distinguishing two pixels if their difference in intensity exceeds 15. Therefore, in general cases for the front regions of cars, the prominent factor threshold $th_{pf}$ is set $2 \times k_{pf} \times 15$. Empirically, the value of $k_{pf}$ is set to 5; the threshold value would be 150. Because the symmetric center and its neighboring pixels in the rear view are not as obvious as those in the front view, the prominent factor threshold of the rear view is generally smaller than that of the front view.

In Fig. 3e, the red curve shows the prominent factors of check points at 27 and 43. Because only the prominent factor of the check point at 43 is large enough, this check point is selected as the symmetric center of the current row. Next, a vote mechanism checks the occurrence frequency of the symmetric centers of all rows. The top 3 frequent symmetric centers are averaged to be the final result.

Through the above procedures, the reference position and the pose angle on the front (or rear) region of a tested car can be extracted. Since the side region is not horizontally symmetric, the symmetric center on the side region cannot be found. An obvious key point in an appearance part, e.g. a corner of a window, can be designated as the reference position. The following procedure introduces the way to acquire the reference position and the pose angle of the side region.

Because the side and front (or the side and rear) regions are segmented from the same car image, the pose angles of the side region and the front (or rear) region must be identical (i.e., $\theta_{test}^{side} = \theta_{test}^{front}$ (or $\theta_{test}^{rear}) = \theta_{test}$). In addition, according to the set of template images for each target car model

used, the distance between the designated reference position and the left end (in quadrant I or III) or the right end (in quadrant II or IV) with known pose angles ($\theta$) can be measured in advance. The ratios of these distances to the width of the car also provide a monotonic function $Ratio_3(\theta)$ for each quadrant. Since the pose angle ($\theta_{test}$) of the side region can be determined, the reference position of the side region can be predicted according to $Ratio_3$ and the position of the left or right end (i.e., $pos_{side} = pos_{left} + Ratio_3(\theta_{test})$ or $pos_{side} = pos_{right} - Ratio_3(\theta_{test})$). Therefore, if we can estimate the pose angle on the front (or rear) region accurately, the reference position and the pose angle on the side region can also be obtained accurately.

As the orientation of the tested car diverges more significantly, it becomes increasingly difficult to locate the symmetric center and to estimate the pose angle. However, the half width of the tested car varies more obviously in large pose angles than in small pose angles. In addition, along with the increment of the orientations, the check width becomes narrower and narrower, so the estimation error of the symmetric center is limited. These effects are capable of balancing the difficulty in extracting the symmetric center and estimating the pose angle in large orientation cases. Therefore, our proposed approach to pose estimation is able to estimate vehicle orientation accurately in most orientation cases.

However, when the orientation is very close to the lateral view (about $90° \pm 10°$ or $270° \pm 10°$), the symmetric center nearly disappears, thereby producing a higher degree of pose estimation error. In this extreme lateral view, the conventional pose estimation method using the ratio $\gamma_1$ is superior because the four ends can be extracted more accurately. A hybrid strategy is employed in the proposed system. When the estimated pose angle utilizing $\gamma_2$ falls within the range of $90° \pm 15°$ (or $270° \pm 15°$) and the pose angle utilizing $\gamma_1$ falls within the range of $90° \pm 10°$ (or $270° \pm 10°$), the estimated pose angle of our proposed approach will be replaced by the estimated pose angle of the conventional approach. This hybrid strategy makes the estimated pose angle more accurate in all orientations.

## 4 Shape feature extraction and segmentation

This section addresses the shape feature extraction and car region segmentation steps in details. When the view of the tested car is angled, the appearances vary significantly in size, shape, and geometric relationship. To correctly determine the locations of the shape features at any orientation becomes critical and challenging. The shape feature extraction algorithm must show flexibility and robustness in its ability adaptive to the geometric deformation. It must also exhibit controllability to prevent from searching in the erroneous

**Fig. 4** ASM result **a** with erroneous initial position, **b** with improper initial mean shape, **c** with proper initialization setting. **d** ASM result of a different model car with the template, **e** the segmented result of **c**, and **f** the segmented result of **d** (color figure online)
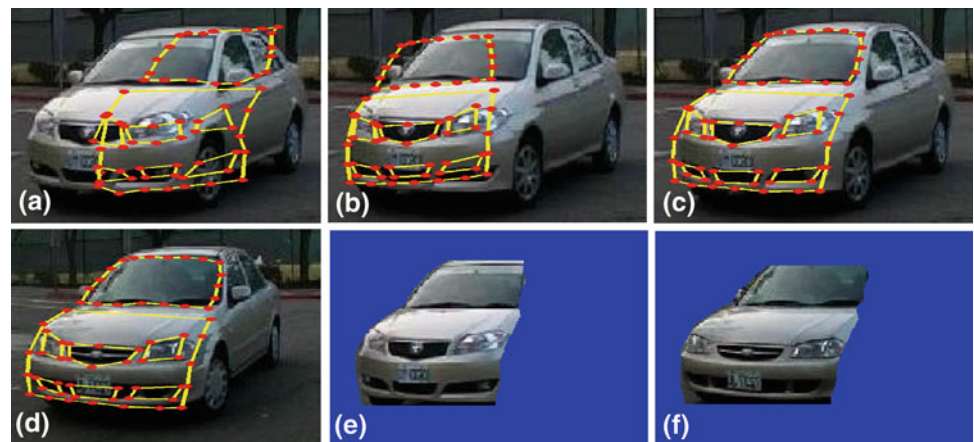


directions. ASM [27] is used to locate key points of the shape feature on cars in this system.

ASM comprises two phases: the off-line training phase and the on-line execution phase. In the off-line phase, several training image sets are prepared for each target model. Each training image set contains 4 or 5 images and covers a specific orientation range. On each training image, a number of key points, such as corners, or boundaries of shape features are marked manually. The coordinates of these key points are collected as a shape vector. Each training set generates a set of shape vectors and computes a mean shape vector. The principal eigenvectors of these shape vectors are calculated to form a geometry variation matrix. The matrix controls the deformation of the mean shape vector by providing the weight for each principal eigenvector. The weight values are varied according to the shape deformation and the deformation is constrained within a suitable limit.

In the on-line phase, ASM deforms the mean shape vector of the selected training image set to search for the corresponding key points on the tested car image iteratively as follows:

$$sv_n = sv_{n-1} + MV \times wv_{ASM}, \qquad (9)$$

where $sv_n$ is the estimated shape vector of the tested car in the $n$th iteration. MV is the geometry variation matrix. $wv_{ASM}$ is the weighting vector.

The initial shape vector $sv_0$ can be represented by two parts: the center position and the orientation of the initial shape vector. The two parts can be acquired by the symmetric center ($pos_{cen}$) and the estimated pose angle ($\theta_2$) extracted by Ratio$_2$ in Sect. 3. After the training set which covers $\theta_2$ is selected, the geometry variation matrix computed from the selected training set is determined. Each key point on $sv_0$ will move toward the closest edge on the tested car image. The movement is constrained by the geometry variation matrix and the weighting vector. Thus, the shape extracted by ASM on the tested car image is constrained to a shape similar to

that of the template. When the ASM searching process converges, the corresponding key points on the tested car image can be located and the shape features of the tested car have been extracted.

Basically, the search process of ASM is sensitive to the initial shape. If the center position is too distant from the correct corresponding position, ASM may search in the wrong direction, so that most key points are incorrectly located. In Fig. 4a, the yellow lines which indicate contours connecting the key points (painted in red) located by ASM, deviate from the actual shape of the tested car. Even though the center position of the initial shape vector is correct, ASM would not guarantee to determine the correct shape of the tested car. In Fig. 4b, if an improper training image set is selected, the orientation of the initial shape vector might quite deviate from the orientation of the actual shape vector. The geometry variation matrix which controls the searching direction may also be erroneous. Thus, to locate the key points of the tested car by ASM more correctly as in Fig. 4c, it is crucial to properly establish the center position of the initial shape vector and the training image set for the ASM searching process. Therefore, an effective approach in the pose and center estimation step is proposed to facilitate the estimation of pose angle and center position for ASM initialization.

Figure 4c, d shows two cases in which we operate ASM for car model recognition. Consider a Toyota VIOS as the target car. In Fig. 4c, the key points on a tested car image of the Toyota VIOS are correctly searched by ASM. In Fig. 4d, a tested car image that is not Toyota VIOS is processed by ASM. ASM also accurately located the corresponding key points on this tested car image without any indication that the two vehicles are not the same model. This implies that utilizing ASM alone has difficulty in distinguishing one particular model of cars from other similar models. Although ASM is a good tool for extracting shape feature from car images, it does not usually perform well on car model recognition. The mirror morphing process described later could help to accomplish the mission.

**Fig. 5** Establish the corresponding relationship between Img$_1$ and Img$_2$ for all pixels based on known key point pairs: **a** source image Img$_1$ with 61 key points, **b** source image Img$_2$ with 61 key points, **c** a pixel $S$ in Img$_1$ but not one of 61 key points, **d** the pixel $S'$ in Img$_2$ corresponding to the pixel $S$ in Img$_1$
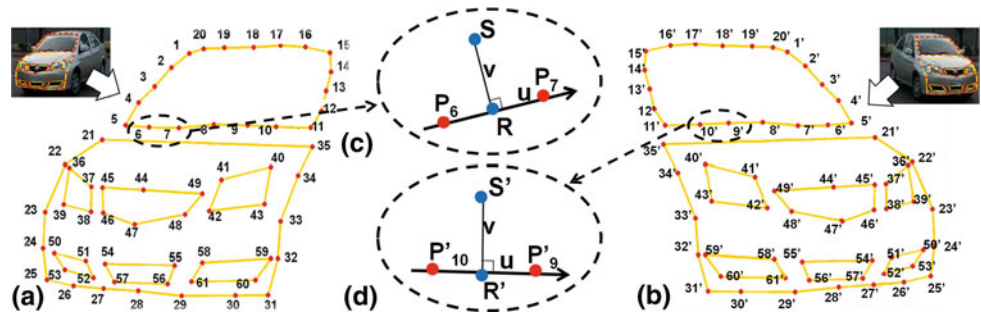


From the ASM results, we separated (segmented) the front regions of cars in Fig. 4c, d and their segmented images are shown in Fig. 4e, f, respectively. It can be seen that the boundaries segmented by ASM are not very smooth and intact. High accuracy cannot be guaranteed using ASM. Although many approaches [37,38] have been proposed to enhance the accuracy of ASM, a state-of-art method has yet to be developed. In our proposed system, the small inaccuracy of ASM can be accommodated by mirror morphing as illustrated in later sections.

## 5 Mirror morphing and matching

This section addresses the mirror morphing step and the template matching step in detail. We extend our previous work [39] by utilizing mirror morphing to transform various poses of a car into a typical (front, rear or side) view. When the pose of the tested car and the template car have both been normalized, the influence of the orientation is significantly reduced and one can expect satisfactory performance in the matching process.

Section 5.1 describes the general morphing technique. Section 5.2 introduces the technique combining general morphing with mirror transform and symmetric correspondence. This technique is termed mirror morphing in the following. Section 5.3 presents the matching process and the recognition performances with and without mirror morphing are compared.

### 5.1 General morphing

The general morphing technique [35] is the process of transforming two source images (Img$_1$, Img$_2$) into a target image (Img). Each pixel in the source images will be transformed (prewarped) to the same coordinate basis. Then, the target image is generated by a weighted combination of the source images controlled by a parameter $\lambda$ as follows.

$$(x_2, y_2) = \text{COR}(x_1, y_1) \tag{10}$$

$$(x, y) = \lambda \times (x_1, y_1) + (1 - \lambda) \times (x_2, y_2) \tag{11}$$

$$\text{Img}(x, y) = \text{Img}_1(x_1, y_1) \times \lambda + \text{Img}_2(x_2, y_2) \times (1 - \lambda), \tag{12}$$

where COR defines the corresponding relationship between a pixel $(x_1, y_1)$ in Img$_1$ and a pixel $(x_2, y_2)$ in Img$_2$. The function Img$(x, y)$ is the intensity at the pixel $(x, y)$ in the target image Img.

The prewarping process requires the correspondence relationship between Img$_1$ and Img$_2$ for all pixels. If we only acquire the correspondence relationship for some key points, not all pixels, the Beier–Neely's Field algorithm [36] can be applied. Figure 5 shows an example to briefly illustrate the Beier–Neely's Field algorithm.

Figure 5a, b shows the key points on the source images: Img$_1$ and Img$_2$. The function ($f$) represents the correspondence between these key points. For example, the key point $P_6$ in Img$_1$ corresponds to the key point $P'_{10}$ in Img$_2$. For a pixel $S$ as shown in Fig. 5c, which is in Img$_1$ but not included in the known key points, we acquire its corresponding pixel $S'$ as shown in Fig. 5d on Img$_2$. The nearest point $R$ to $S$ along line segment $P_m P_n$ ($m = 6, n = 7$ in Fig. 5a) is obtained. The distance from $S$ to $R$ is denoted by $v$ and the distance from $P_m$ to $P_n$ is denoted by $u$. These distances can be computed by Eq. (13). Function Perp returns the vector perpendicular to the input vector. Because $S'$ corresponds to $S$, $S'$ will be located at $v$ pixels away from the line segment $P'_{f(m)} P'_{f(n)}$ ($f(m) = 10, f(n) = 9$ in Fig. 5b) as $S$ away from the line segment $P_m P_n$. Hence, $S'$ in Img$_2$ can be computed by Eq. (14).

$$u = \frac{(S - P_m) \cdot (P_n - P_m)}{\|P_n - P_m\|^2}, \quad v = \frac{(S - P_m) \cdot \text{Perp}(P_n - P_m)}{\|P_n - P_m\|^2} \tag{13}$$

$$S' = P'_{f(m)} + u \cdot (P'_{f(n)} - P'_{f(m)}) + \frac{v \cdot \text{Perp}(P'_{f(n)} - P'_{f(m)})}{\left\|P'_{f(n)} - P'_{f(m)}\right\|^2}. \tag{14}$$

When multiple pairs of line segments are considered, the weight of each corresponding line segment needs to be calculated. The weight is determined by the distance from $S$ to each line segment $P_m P_n$. The closer the pixel to the line segment, the higher the weight is.

### 5.2 Mirror morphing

Next, we utilize a mirror morphing technique to synthesize a segmented car image on the front (or rear) region into

**Fig. 6** **a** Segmented image of front region, **b** mirror image of **a**, **c** synthesized car image from **a** and **b** by mirror morphing, **d** synthesized template car image



an exactly front (or rear) view image. Since the horizontal symmetry does not exist in the side region, a quasi mirror morphing is designed. The quasi mirror morphing is able to transform the segmented car image on the side region into an approximate typical side view image. The mirror morphing and the quasi mirror morphing are introduced as follows in order.

Because a car is generally symmetric horizontally in the front (or rear) region, for any tested car image $Img_1$ segmented from the front (or rear) region, an opposite view image $Img_2$ can be produced by mirror transformation as follows.

$$(x_2, y_2)_i = (Width(Img_1) - x_1, y_1)_i \tag{15}$$

$$Img_2(x_2, y_2)_i = Img_1(Width(Img_1) - x_1, y_1)_i, \tag{16}$$

where $Width(Img_1)$ represents the width of $Img_1$. The opposite view image termed the mirror image is adopted as the other source image $Img_2$ for morphing.

Since $Img_2$ is produced by $Img_1$, the correspondence relationship between the key points in $Img_1$ and $Img_2$ can be acquired if the shape vector is defined to be symmetric. For example, if $Img_2$ in Fig. 5b is the mirror image of $Img_1$ in Fig. 5a, the key points $(P_{36}-P_{39})$ on the left car lamp in Fig. 5a are transformed into the key points $(P'_{36}-P'_{39})$ on the right car lamp in Fig. 5b. The transformed right car lamp $(P'_{36}-P'_{39})$ in $Img_2$ should be morphed with the right car lamp $(P_{40}-P_{43})$ in $Img_1$. Therefore, if the correspondence relationship between the key points $(P_{36}-P_{39})$ and $(P_{40}-P_{43})$ in $Img_1$ has been known, the correspondence relationship between the key points $(P_{40}-P_{43})$ in $Img_1$ and $(P'_{36}-P'_{39})$ in $Img_2$ is acquired.

We define the shape vector (i.e., landmark the locations of key points) of ASM to be symmetric. Hence, each key point $(x_1, y_1)_i$ in $Img_1$ has its corresponding key point $(x_1, y_1)_{Cor(i)}$ in the opposite side in $Img_1$. For example, $P_1$ corresponds to $P_{15}$ and $P_5$ corresponds to $P_{11}$ in Fig. 5a. Let $(i, j)$ denote the index of key points to be the corresponding pair, a function Cor is denoted as Eq. (17).

$$j = Cor(i). \tag{17}$$

According to Eqs. (15) and (16), all key points in the mirror image $Img_2$ is generated by the key points in the image $Img_1$. By knowing the corresponding pairs of the key points in $Img_1$, the correspondence between the key point

$(x_2, y_2)_i$ in $Img_2$ and the key point $(x_1, y_1)_{Cor(i)}$ in $Img_1$ can be acquired. For example, $P'_1$ in Fig. 5b is generated by $P_1$(i.e., $P'_1 = Width(Img_1) - P_1$) and $P_1$ corresponds to $P_{15}$ in Fig. 5a, so $P'_1$ corresponds to $P_{15}$. After the correspondences between all the key points in $Img_1$ and $Img_2$ are known, the Beier–Neely's Field algorithm [36] can be utilized to extend the correspondences from these key points to all pixels. Finally, the synthesized image can be produced as follows:

$$
\begin{aligned}
(x, y)_{Cor(i)} &= \lambda \times (x_1, y_1)_{Cor(i)} + (1 - \lambda) \times (x_2, y_2)_i \\
&= \lambda \times (x_1, y_1)_{Cor(i)} + (1 - \lambda) \\
&\quad \times (Width(Img_1) - x_1, y_1)_i
\end{aligned} \tag{18}
$$

$$
\begin{aligned}
Img(x, y)_{Cor(i)} &= \lambda \times Img_1(x_1, y_1)_{Cor(i)} + (1 - \lambda) \\
&\quad \times Img_2(x_2, y_2)_i \\
&= \lambda \times Img_1(x_1, y_1)_{Cor(i)} + (1 - \lambda) \\
&\quad \times Img_2(Width(Img_1) - x_1, y_1)_i.
\end{aligned} \tag{19}
$$

Figure 6a is the same segmented car image of Fig. 4e, and Fig. 6b is the mirror image of Fig. 6a. We construct the correspondence relationship between Fig. 6a and b by their ASM results and synthesize Fig. 6a, b by the general morphing process [35] into Fig. 6c to complete mirror morphing. Since mirror morphing is able to transform the tested car image into its typical front (or rear) view, the pose variation and center bias are eliminated. Although the error of locating the shape features of the tested car may make the synthesized image a little blurred, the blurred phenomenon will not degrade the recognition performance as much as the effects of pose error and center bias. The performances with and without mirror morphing will be compared in Sect. 5.3.

To synthesize the segmented image of the side region into an exactly lateral view (90° or 270°), a mirror image orientated to the opposite lateral view is required. However, this mirror image cannot be generated using mirror transformation because horizontal symmetry does not exist in the side region. In this study, if the estimated pose angle of the tested car is $\theta^{test}$, a template image with pose angle $180° - \theta^{test}$ (or $540° - \theta^{test}$) is selected as the mirror image. For example, because the estimated pose angle of the tested car in Fig. 7a is 70°, the template with the pose

**Fig. 7** **a** Segmented image of side region, **b** selected mirror template, **c** synthesized car image from **a** and **b** by quasi mirror morphing



angle of 110° shown in Fig. 7b is selected to be the mirror image.

Since the shape features of the segmented image on the side region and its mirror image can be extracted by ASM in the shape feature extraction step, the shape features referring to the same characteristics are in correspondence. Then, the general morphing process is performed on the segmented side region image (e.g. Fig. 7a) and the selected template image (e.g. Fig. 7b). An approximate lateral view image is synthesized as shown in Fig. 7c. This synthesizing procedure for the side region image is termed as the quasi mirror morphing process.

Since the mirror image is not produced from the segmented side region of the tested car, the quasi mirror morphing process can reduce but not eliminate the pose error as the mirror morphing does in the front or rear region. Fortunately, shape deformations in situations close to the lateral view are generally less serious than those found in other views. Assume $\Delta\theta$ is the variation of pose angle of a tested car. In the lateral view, deformations are mainly due to the width of the car multiplying $\sin(\Delta\theta)$ and the length of the car multiplying $1 - \cos(\Delta\theta)$. While, in the front or rear view, the deformations are mainly due to the width multiplying $1-\cos(\Delta\theta)$ and the length multiplying $\sin(\Delta\theta)$. Since $\sin(\Delta\theta)$ is obviously larger than $1 - \cos(\Delta\theta)$ when $\Delta\theta$ is small and the car length is much larger than the car width, the shape deformations of cars in the lateral view would be smaller than those in the front or rear view. This effect could increase the tolerance on pose error in the neighborhood of the lateral view when we recognize car models by template matching. Hence, the quasi mirror morphing process can still provide good recognition performance in cases close to the lateral view.

### 5.3 Template matching

The morphed tested car image (Fig. 6c) is matched against the morphed template car image (Fig. 6d) to recognize the car model. The morphed template with the orientation closest to the estimated pose angle of the tested car is selected from the template database. The sum of absolute differences (SAD) method which evaluates the SAD score is used for template matching. The SAD score is defined as the SAD of r, g, and b channels for each pixel between the tested car and the template car, and averages over all pixels on the tested car. The SAD score is referred to the matching error of the tested car image. If the matching error is smaller than a predefined

threshold $th_{model}$, the tested car and the template car are considered to be the same model; otherwise, they are considered to be different model cars.

In multi-model recognition, $N$ morphed templates with the orientations corresponding to the estimated pose angles of the tested car for the $N$ target car models are selected. The morphed tested car image is then matched against all the selected morphed templates sequentially. For each target car model, the matching error between its morphed template and the morphed tested car image is computed. Then, the matching error is compared with a predefined threshold which is trained for each target car model. If the matching error is larger than the predefined threshold, the tested car will not be identified as the same model of that target car model. Among the target car models whose matching errors are smaller than their thresholds, the target car model whose matching error is most distinct to its threshold is selected to be the model of the tested car.

The threshold $th_{model}$ is decided by the number of non-target model cars that tolerant to misclassification as the target model car. The more the misclassified non-target model cars are tolerated, the larger the threshold is. Thus, the tested cars are easier to be identified as the target model. The percentage of misclassified non-target model cars is empirically set to a very small value, such as 0.3%.

We consider the scenario that the cars with the same shape but different colors to be different models, so the tested car which has a different color to that of the template car will have a large SAD score and will not be identified as the target model. The SAD score is calculated after intensity normalization. The intensity normalization contains computing the average intensity of the template car first and then adjusting the average intensity of the tested car to the average intensity of the template car. After intensity normalization, the SAD method is able to accommodate general lighting and weather variations.

We compared the SAD method with two well-known template matching methods, the normalized cross correlation (NCC) and the edge-based techniques [40]. The NCC method can provide slightly superior recognition accuracy to the SAD method, but the computation load of NCC is much heavier than the computation load of SAD. The edge-based method can eliminate the influence of vehicle colors, while the improvement from the edge-based method in the scenario that we are investigating is quite limited. Furthermore, the edge-based method usually loses some texture information

**Fig. 8** Sensitivities of the algorithms with and without mirror morphing to pose error and center bias in the initial phase
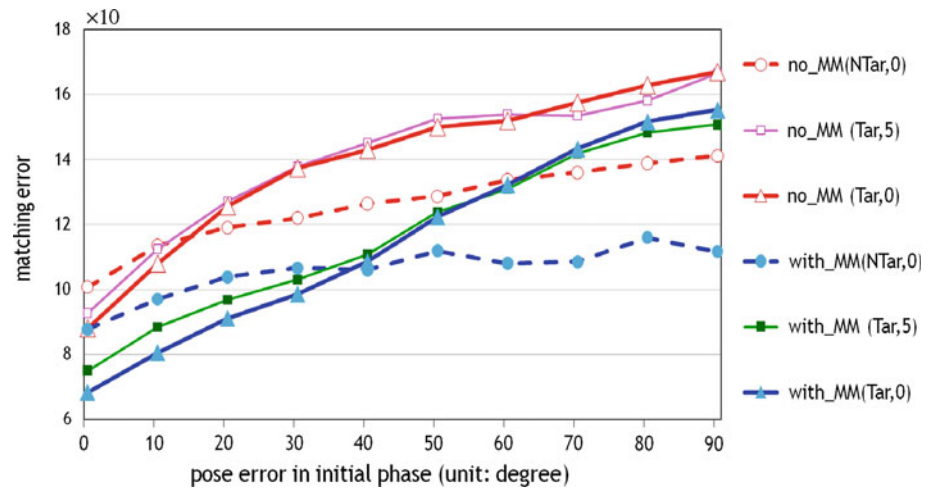


and generates significant errors from falsely detected edges. Generally, the edge-based method obtains worse car recognition rate than the SAD method. Since the SAD method shows robust performance and efficient computation, the SAD method is adopted in the template matching step of the proposed system.

Figure 8 illustrates the matching errors by algorithms with and without mirror morphing (with_MM/ no_MM) with respect to various pose estimation errors (from 0° to 90°) and center biases (0 and 5 pixels) in the initial phase. The tested cars that are the same models as the templates are referred to as target model cars (Tar). The tested cars that are different models to the templates are called non-target model cars (NTar). When the algorithm is tested on target model cars, the matching errors shown in Fig. 8 are the statistical results of numerous tested car images. Each tested car image has an SAD score. The root mean square (RMS) value of all the computed SAD scores with the same estimated pose error in the initial phase is calculated as the statistical result. Since non-target model cars are utilized to determine the threshold for recognition, the matching error associated with the non-target model cars in Fig. 8 shows the minimum SAD over all non-target car images. This minimum value can be considered the threshold. The target model cars with their SAD scores lower than the threshold are considered correct recognitions. Therefore, as shown in Fig. 8, if the (RMS) matching error of an algorithm for all target model cars is obviously smaller than the minimum SAD for the non-target model cars, the performance of the algorithm is considered good.

As seen in Fig. 8, even in ideal conditions with no center bias, the algorithm without mirror morphing (no_MM(Tar,0)) provided only moderate performance when the pose estimation error is less than 10°. In conditions in which the center bias is 5 pixels, the algorithm without mirror morphing (no_MM(Tar,5)) achieves only moderate performance when the pose estimation error is smaller than 3°. Such

requirements are extremely difficult to achieve. However, when the center bias is zero, the algorithm with mirror morphing (with_MM(Tar,0)) can work well when the pose estimation error is smaller than 30°, and in cases in which the center bias is 5 pixels, the algorithm with mirror morphing (with_MM(Tar,5)) still works well even when the pose estimation error is less than 20°. Such requirements can be easily achieved in most practical situations. It can be concluded that mirror morphing is an effective technique for the recognition of car models and other applications dealing with severe variations in pose.

## 6 Experimental results

The experiments were performed and compared to demonstrate the superiority of our proposed ideas. In Sect. 6.1, several algorithms are introduced for comparison. Section 6.2 describes a car image database and the metrics to evaluate the recognition performance. The performance and efficiency of the proposed system are demonstrated in Sect. 6.3.

6.1 Algorithms for comparison

Three algorithms: PETM, PASM, and ASMM are presented to demonstrate that the proposed mirror morphing technique provides a considerably improvement to the conventional pose estimation and matching methods. Subsequently, a different type of algorithm, QRAFM [20] is implemented and compared with the proposed algorithm, ASMM. These algorithms are described as follows:

(1) PETM (Pose estimation and template matching)
PETM performs the pose and center estimation first and then selects a template car image with the orientation closest to the estimated angle. All the tested car images are offset to the center position of the selected

template and then matched against the selected template. PETM can represent the category of a baseline car model recognition method [15] utilizing pose estimation and template matching.

(2) PASM (Pose estimation and ASM)

PASM performs ASM to extract shape features after the pose angle and center position have been estimated. The center position will be re-estimated by the center of the shape features, and a better alignment for template matching is acquired. The segmentation processing is also performed and the tested car image is matched against the template with the orientation closest to the estimated pose angle. PASM can represent the category of an advanced car model recognition method [19] using pose estimation and template matching.

(3) ASMM (ASM and mirror morphing)

ASMM is our proposed algorithm which performs the whole set of functions of five steps as depicted in Fig. 1. Comparing with the algorithm PASM, mirror morphing is performed for both tested and template car images. Then, the morphed tested car image is matched against the morphed template car image with the orientation closest to the estimated pose angle. In PASM and ASMM, the front (or rear) view and the side view images of the tested car are extracted separately and they are matched against their individual templates.

(4) QRAFM (Quasi-rigid alignment and flexible matching)

The QRAFM algorithm [20] proposes a quasi-rigid alignment technique and designs a local flexible matching mechanism to recognize cars. The tested car is aligned to the template car based on the established corresponding relationships between the key segments on the tested car and the template car. Subsequently, the local flexible matching mechanism divides the tested car into multiple patches. Each patch is shifted in the neighboring regions to obtain the optimal match to the corresponding part of the template car.

Since the procedures of PETM, PASM and ASMM are similar, more detailed analyses are provided to demonstrate the contribution when the mirror morphing technique is involved. Because the accuracies of the estimated pose angle and center position considerably affects the performance comparison, two estimation approaches, introduced in Sect. 3, are tested. The conventional pose estimation approach based on the ratio of car width to car height is denoted as A, and our proposed pose estimation approach based on the ratio of half front (or rear) width to car width is denoted as B. The PETM, PASM, and ASMM algorithms can use either A or B as the initial setting to analyze the intermediate data. The names of these algorithms and their respective functionality are illustrated in Table 1. Because the QRAFM algorithm

**Table 1** The functions working in the PETM, PASM, and ASMM algorithms

| Step Algo. name | (1) Pose & center estimation | (2) Shape feature extraction | (3) Car region segmentation | (4) Mirror morphing | (5) Template matching |
|---|---|---|---|---|---|
| PETM-A | △ | × | × | × | ○ |
| PETM-B | ◆ | × | × | × | ○ |
| PASM-A | △ | ○ | ○ | × | ○ |
| PASM-B | ◆ | ○ | ○ | × | ○ |
| ASMM-A | △ | ○ | ○ | ○ | ○ |
| ASMM-B | ◆ | ○ | ○ | ○ | ○ |

△ the conventional pose estimation approach, ◆ our proposed pose estimation approach, × the step is not performed, ○ the step is performed

adopts different strategies for recognition, the initial setting is not applied and only the recognition performance is compared.

6.2 Experimental description

In the experiments, we compare the performance of the algorithms in car model recognition. To distinguish a car from other cars with distinctly different shape or color is generally not a difficult problem and can be accomplished by ASM or the template matching techniques easily. In this paper, we focus on the challenging cases that recognize cars with similar shape and color.

A car image database [41] is constructed containing over 6,000 car images with orientations from 0° to 360°. The images were captured in real traffic streams and some examples are shown in Table 2. The images in the database are categorized as target model and non-target model. Two car models, Toyota VIOS in silver and Nissan New SENTRA in yellow, are selected as the target car models. All the non-target model cars are similar in shape and color to the target model cars.

In the dataset, 36 car images at 0°, 10°, 20°, 30°, ... and 350° for each target model are collected as the template images. In the pose and center estimation step, the template images are divided into four sets, [0°, 90°], [90°, 180°], [180°, 270°], and [270°, 360°]. In each set, ten template images are utilized to construct a monotonic function to estimate the pose angle of a tested car. In the shape feature extraction step, ASM requires 36 training sets to train the ASMs for all sample pose angles ($\theta = 0°, 10°, ..., 350°$). Each training set collects the required template images as a set, covering the following pose angles: $\theta - 20°, \theta - 10°, \theta, \theta + 10°$, and $\theta + 20°$ for the sample pose angle $\theta$. In the mirror morphing step, the side regions of all 36 template images are segmented for quasi-mirror morphing. In the template matching step, all 36 template images are morphed and one of

**Table 2** Examples of database images. Each row displays car images in the same view. The first car with red boundary is the target model and the others are non-target models (color figure online)



| Target model (Toyota VIOS) and Non-target model cars in similar shape and color | Target model (Nissan NEW SENTRA) and Non-target model cars in similar shape and color |
|---|---|
| *Front (θ = 0˚)* | *Front (θ = 0˚)* |
| *Profile (θ = 45˚)* | *Profile (θ = 45˚)* |
| *Side (θ = 90˚)* | *Side (θ = 90˚)* |
| *Profile (θ = 135˚)* | *Profile (θ = 135˚)* |
| *Rear (θ = 180˚)* | *Rear (θ = 180˚)* |

the morphed template images with the orientation closest to the estimated pose angle of the tested car is selected for matching.

All images of non-target model cars were tested first when the experiment begins. The system pre-assumed the tested cars to be the target model, and matched them against the target car template. The matching error between the tested car and the target car template was computed. Cars that were non-target models with low matching error for misclassification as the target model were called false alarms. To determine whether a tested car was target model or not, a threshold was defined to maintain the probability of false alarms as Eq. (20) at an acceptable level. For fair comparison, the probability of false alarm ($P_{fa}$) was maintained the same for all algorithms and the respective threshold is set thereby.

$$P_{fa} = \frac{\text{\# of non-target model misclassified as the target model}}{\text{\# of non-target model}} \quad (20)$$

When the threshold was set properly, we next matched the target model cars against the target car template under the same $P_{fa}$ for all algorithms. The target model cars whose matching error is smaller than the threshold are correctly classified as the target model. The rate of successful recognition

($P_{rec}$) of the target model cars was calculated, as illustrated in Eq. (21).

$$P_{rec} = \frac{\text{\# of target model correctly classified as the target model}}{\text{\# of target model}} \quad (21)$$

### 6.3 Performance evaluation

The following performance indices are evaluated for the PETM, PASM, and ASMM algorithms: (1) accuracy of pose angle and center position estimation, (2) similarity of target and non-target model cars, (3) recognition rate over all orientations. Only the similarity index and recognition rate of QRAFM are measured because the QRAFM algorithm does not estimate the pose of a tested car. Finally, (4) the computation load of each functional step is demonstrated to verify the efficiency of the proposed system.

#### 6.3.1 Accuracy of pose angle and center position estimation

The accuracies of pose angle and center position estimations will affect the performance of ASM and mirror morphing, therefore the recognition performance is also influenced significantly. For each tested car, the ground truth of pose

**Fig. 9** Accuracy of estimated pose angle over all orientations of the six algorithms
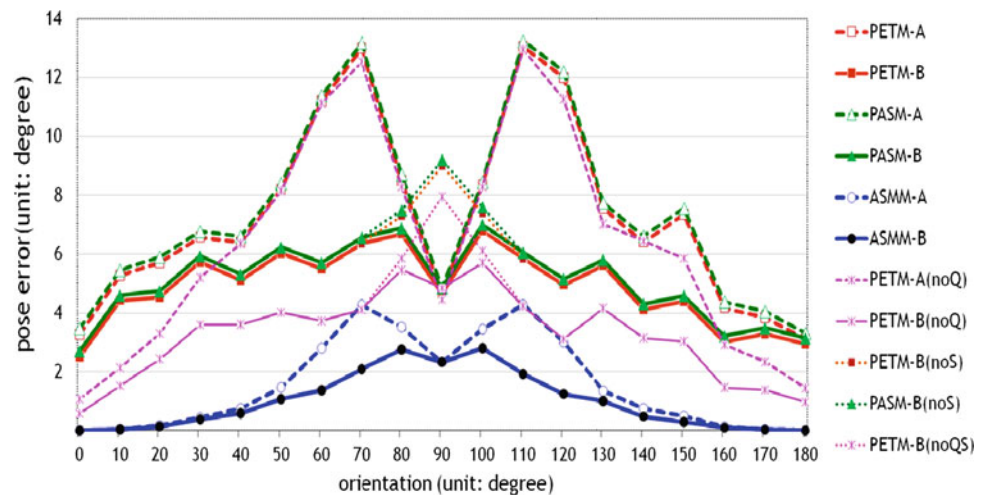


**Fig. 10** Accuracy of estimated center position over all orientations of the six algorithms
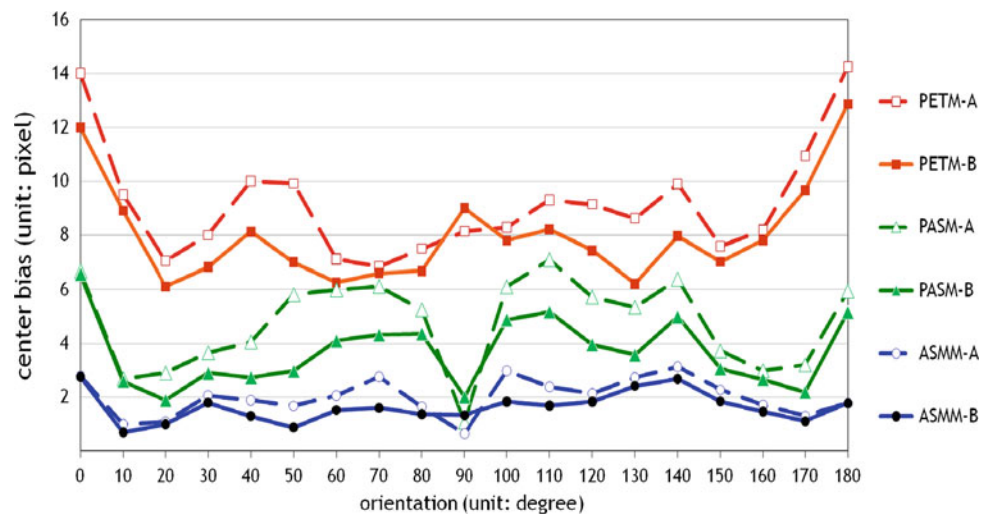


angle is manually defined. After the pose angle of the tested car is estimated by a pose estimation approach, either A or B, a template car that covers the estimated angle is selected. The absolute difference between the ground truth and the estimated pose is defined as the pose error without quantization (noQ). The absolute difference between the ground truth and the pose angle of the template is defined as the pose error with quantization. Because we took 10° as the angle unit for preparing the templates, the pose error within ±5° will exist inherently for each algorithm when the tested car is matching against the selected template.

Figures 9 and 10 show the statistical accuracies (the RMS errors over numerous tested car images) of the estimated pose angle and center position for the six algorithms: PETM-A, PETM-B, PASM-A, PASM-B, ASMM-A, and ASMM-B. In Fig. 9, the pose errors without quantization are also shown as PETM-A(noQ) and PETM-B(noQ). In our experiments, the quadrants of the orientations for all target model ca rs can be correctly identified. Due to the symmetric property, only the

performances of the half orientations [0°, 180°] are shown in the figures.

In Fig. 9, the pose errors in PETM-A are about from 3.27° to 13.02°. Larger errors appear when the orientations are within [40°, 80°] and [100°, 150°]. The algorithm PETM-B always has smaller pose errors (from 2.50° to 6.49°) than PETM-A, especially when the orientation is within [40°, 80°] and [100°, 150°]. This indicates that our proposed pose estimation approach (B) provides smaller and stable pose error than the conventional pose estimation approach (A) over most orientations.

However, PETM-A exhibits smaller pose errors in the neighborhood of the lateral view [80°, 100°] than in other views. While, the PETM-B without substitution processing (denoted by PETM-B(noS) in Fig. 9) demonstrates larger pose error in these views [80°, 100°]. As mentioned in Sect. 3, a hybrid strategy (with substitution processing) is employed to improve the accuracy of PETM-B in cases close to the lateral view. Therefore, PETM-B can provide accurate pose angle estimation over all orientations.

**Fig. 11** Matching error distribution of **a** PETM-A and **b** PETM-B on target cars and non-target cars (color figure online)
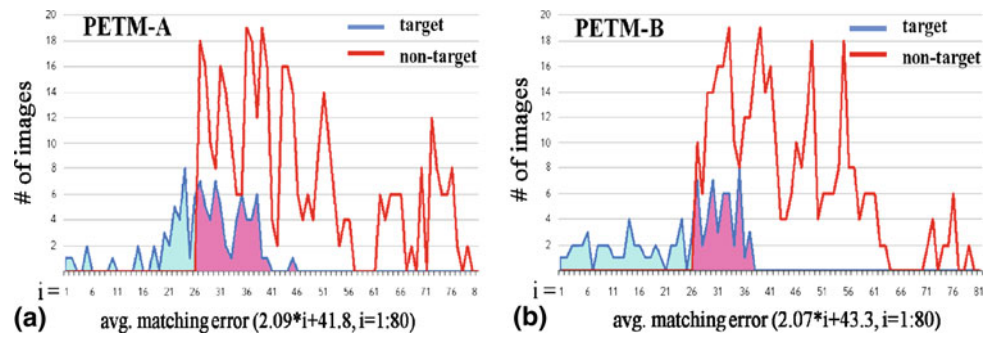


**Fig. 12** Matching error distribution of **a** PASM-A and **b** PASM-B on target cars and non-target cars (color figure online)
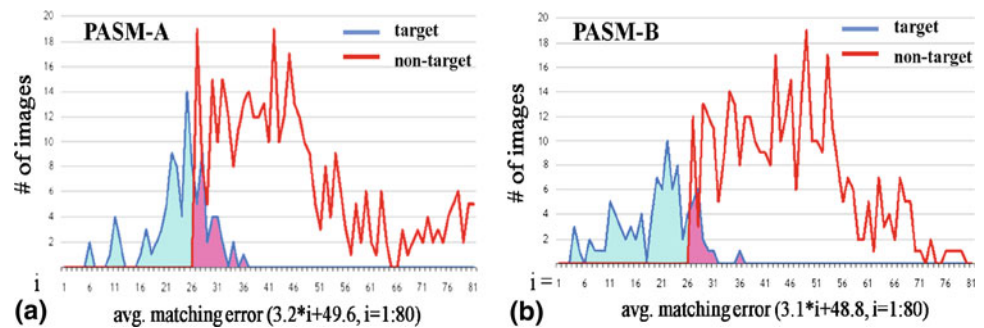


Since the pose errors of PETM-B and PETM-B(noQ) are always smaller than 7° and 5°, respectively, the proposed pose estimation approach (B) would be sufficient to support the ASM initialization, mirror morphing and template matching. PASM adopts the estimated pose angle obtained in PETM, so the pose errors of PASM-A and PASM-B are the same as those in PETM-A and PETM-B.

Figure 10 shows the center bias of the six algorithms. Because the pose error of PETM-B is smaller than PETM-A, the center bias of PETM-B is also better than that in PETM-A. Excluding the cases of very small orientation, the center bias is from 6.12 to 8.13 pixels in PETM-B and from 6.85 to 10.00 pixels in PETM-A. Because PASM-A and PASM-B can get more location information by ASM than PETM-A and PETM-B, a more accurate center position will be re-estimated for PASM-A and PASM-B. Within the orientation [0°, 50°] and [140°, 180°], the center bias in PETM-A is from 7.05 to 14.01 pixels and from 6.12 to 12.00 pixels in PETM-B. Whereas, the center bias is only from 2.69 to 6.68 pixels in PASM-A and is from 1.89 to 6.58 pixels in PASM-B. Within the orientation [50°, 80°] and [100°, 140°], PASM still has more accurate estimation of center position, but the improvement is not so significant as in the cases of pose angle [0°, 50°] and [140°, 180°].

Even the algorithm PASM-B remains small estimation errors of pose angle and center position, the recognition performance is still influenced dramatically by the small errors. To eliminate these errors, the ASMM algorithm adopting mirror morphing is proposed. The ASMM-A and ASMM-B algorithms normalize both the poses of the tested car and the template car to an exactly typical (front or rear) or

approximately typical (side) view. In Fig. 9, the pose errors using our proposed algorithms, ASMM-A and ASMM-B, are approximately zero when the orientation is close to 0° or 180°. When the orientation approaches to the lateral view [80°, 100°], the quasi mirror morphing process is dominant to work for the side region, thereby increasing pose error. In ASMM-A, the maximum pose error is 4.12°, appearing at 70° and 110°. In ASMM-B, the maximum pose error is 2.54° appearing at 80° and 100°. Since the shape deformation is less serious in the neighborhood of the lateral view, this degree of pose error can be tolerated and a satisfactory recognition rate can still be obtained, as introduced later in Fig. 15.

The errors of center (reference) position bias in the horizontal axis can also be eliminated by mirror morphing and the remaining vertical center bias in ASMM-A and ASMM-B shown is small. In Fig. 10, if excluding the cases of very small orientation, the maximum center bias is 2.76 pixels in ASMM-A and 2.23 pixels in ASMM-B. Since the ASMM-A and ASMM-B algorithms have very small pose error and center bias, very good recognition performance can be obtained.

### 6.3.2 Similarity of target and non-target model cars

The similarity of the target model cars and the non-target model cars determines whether they are easily to be distinguished or not. A good recognition algorithm should make the target model cars and those of the non-target model cars have very low similarity even they are in the severe pose variation condition and similar in shape and color. Figures 11, 12, 13 and 14 show the distribution of the matching errors

Fig. 13 Matching error distribution of **a** ASMM-A and **b** ASMM-B on target cars and non-target cars (color figure online)
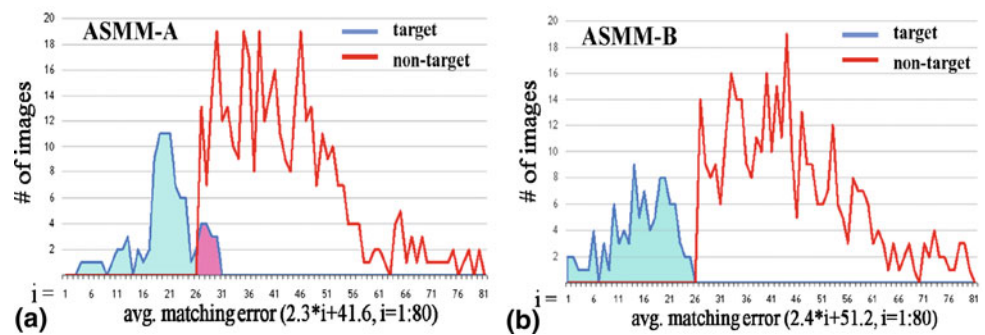
Fig. 14 Matching error distribution of QRAFM adopting the template car whose orientation is **a** 30°, and **b** 50° on target cars and non-target cars (color figure online)
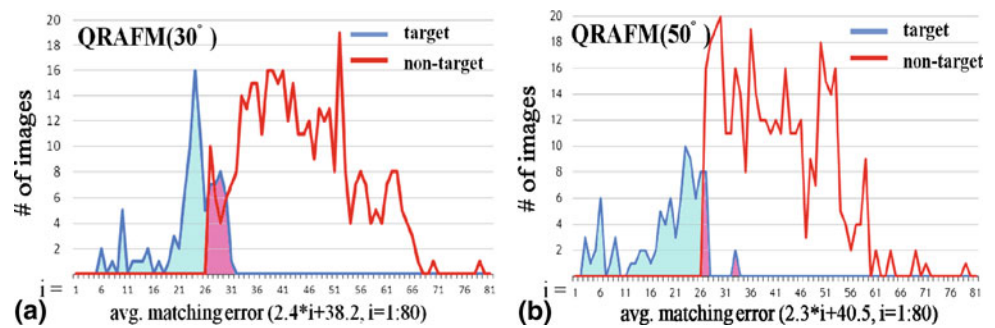
(SAD-score) between the tested car images to the template image.

In Figs. 11, 12, 13 and 14, the matching errors of the target and non-target model cars fully spread over the horizontal axis are scaled in a range from 1 to 80. The matching error is represented by the value of the symbol "$i$" and the equation $a \times i + b$ below the horizontal axis. The symbol "$i$" represents the value on the horizontal axis ($i = 1, \ldots, 80$). The matching error is computed by substituting the value of "$i$" into the equation. For example, when the value of "$i$" is 26, the value of matching error in Fig. 11a is $2.09 \times 26 + 41.8 = 96.14$.

The red and blue curves in Figs. 11, 12, 13 and 14 show the distributions of the matching errors of the non-target cars and the target cars with the template car in the cases of orientation $50° \pm 5°$. The red curve can be served as the basis to establish a specific threshold (th$_{model}$) for controlling the probability of false alarm ($P_{fa}$). The blue curve can be utilized to evaluate the successful recognition rate ($P_{rec}$) under the specified threshold. Only the target model cars whose matching errors are below the threshold are correct recognitions. Therefore, the area painted by the blue color shows the cases of successful recognition, while the overlap area painted by the pink color shows the cases of fail recognition. The size of the pink area represents the similarity of target cars and non-target cars on each algorithm and the size of the blue area displays the recognition performance.

From Fig. 11a, in the algorithm PETM-A, the pink area occupies 60.42% of the target model cars area. Thus, the recognition rate of PETM-A is 39.58%. From Fig. 11b, in the algorithm PETM-B, the pink area occupies 52.69% of the target model cars area. Thus, the recognition rate of PETM-B

is 47.31%. Our proposed pose estimation approach provides about 8% improvement from PETM-A to PETM-B.

Figure 12a, b shows the overlapped (pink) area of target and non-target model cars of PASM-A and PASM-B, respectively. Since the algorithm PASM utilizes ASM to get more accurate center position than PETM, PASM has obviously smaller pink areas (30.21% in PASM-A and 18.75% in PASM-B) than PETM. Thus, the recognition rate of PASM-A and PASM-B would be 69.79 and 81.25%, respectively.

Because of the benefit of mirror morphing, the algorithms ASMM (including ASMM-A and ASMM-B) show obviously smaller pink area than the above mentioned algorithms. In Fig. 13a, the percentage of the pink area in ASMM-A is 13.54%, so that the recognition rate is 86.46%. In ASMM-B, because our proposed pose estimation approach (B) provides more accurate initial setting for ASM than the conventional approach (A), the shape feature of the tested car can be extracted by ASM more accurately. From Fig. 13b, the matching error distribution of the target model cars and the non-target model cars in ASMM-B is almost completely disjoint, so that the pink area almost disappears (0%) and the recognition rate is nearly 100%.

Figure 14a, b shows the distribution of the matching errors of the QRAFM algorithm when the tested cars with orientations of $50° \pm 5°$ are aligned to the template cars with orientations of 30° and 50°, respectively. The alignment may deform the shapes of the tested cars. The greater the pose differences between the tested cars and the template car, the more serious the deformation will be. In Fig. 14a, because of larger pose differences (approximately 20°) between the tested cars and the template car, the percentage of the pink area is 29.63%

**Fig. 15** Recognition rate over all orientations of the PETM, PASM and ASMM algorithms
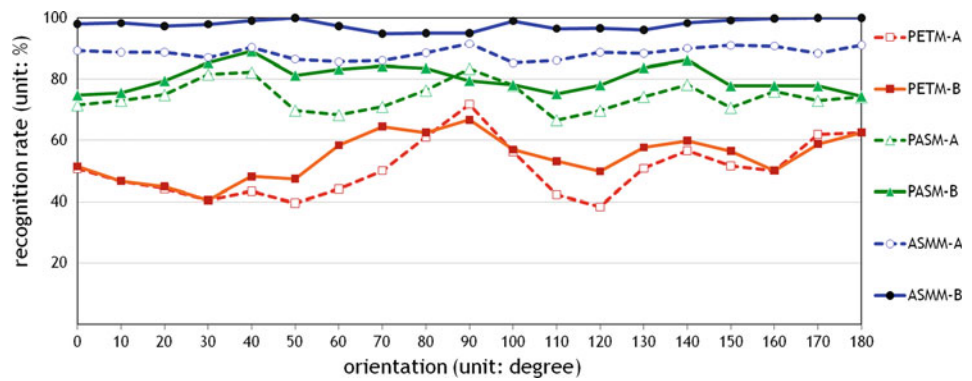


**Fig. 16** Recognition rate over all orientations of the ASMM-B, PE-QRAFM, QRAFM with 45° and 135° templates, and PASM-B algorithms
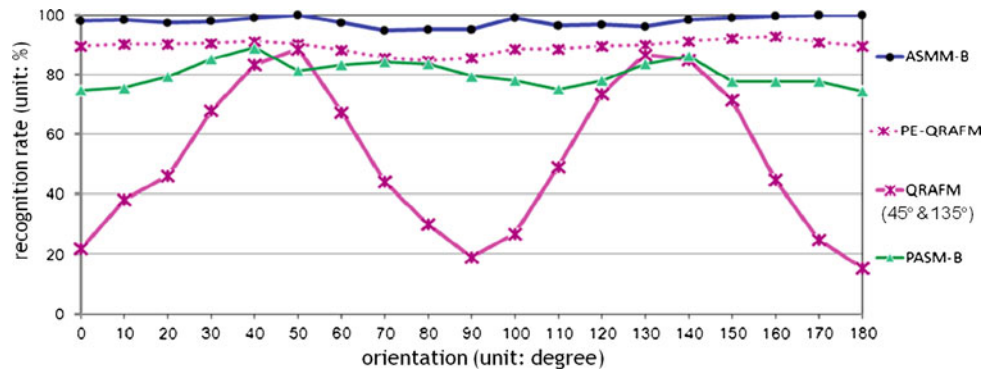


and the recognition rate is 70.37%. In Fig. 14b, the absolute pose differences between the tested cars and the template car are smaller than 5°. Hence, the pink area occupies 10.43% of the target cars, therefore the recognition rate of the QRAFM algorithm achieves 89.57%.

By comparing ASMM with QRAFM, the recognition rate of ASMM (Fig. 13b) is considerably higher than that of QRAFM (Fig. 14a or b). In Fig. 14b, the orientations of the tested cars are close to the orientation of the template car, but ASMM is still superior to QRAFM. The QRAFM algorithm will align the tested cars including target and non-target cars to the same shape with the template car, whereas ASMM morphs the tested cars depending on their original images and mirror images. Hence, ASMM demonstrates a lower degree of similarity between the target and non-target cars than that provided by QRAFM.

### 6.3.3 Recognition rate over all orientations

We show the recognition rates of all algorithms for the tested cars over all orientations in Figs. 15 and 16. Due to the symmetric property, only the performances of the half orientations (0°–180°) are shown.

In Fig. 15, the algorithm PETM-A has a moderate recognition rate from 40 to 70% for the entire orientations. The recognition rate of PETM-B is improved about 5–15% than those in PETM-A when the orientation is within [40°, 70°] and [110°, 140°]. The recognition rate of PASM-B can also

be improved about 5–15% than those of PASM-A within the same orientation range. The improvement is because our proposed pose estimation approach (B) provides smaller pose error than the conventional pose estimation approach (A) when the pose angle is within [40°, 80°] and [100°, 150°] as shown in Fig. 9.

Because the PASM algorithms (including PASM-A and PASM-B) can reduce the errors of center position significantly as shown in Fig. 10, the recognition performance can be highly enhanced. In Fig. 15, the recognition rate of PASM-A achieves 65–80% and the recognition rate of PASM-B achieves 75–90%. These recognition performances are much better than those of PETM-A and PETM-B and the improvement is about 10–30%.

Since mirror morphing can eliminate the estimation errors of pose angle and center position, the algorithms ASMM-A and ASMM-B can obtain very good recognition performance. In Fig. 15, ASMM-A achieves the recognition rate from 85 to 90% and the recognition rate of ASMM-B exceeds 95% over all orientations. ASMM-B is superior to ASMM-A because our proposed pose estimation approach (B) can provide more accurate pose and center estimations. The shape features of cars will be extracted more accurately and the effect of the blurred phenomenon through mirror morphing is alleviated.

In the neighborhood of the lateral view [80°, 100°], since the conventional pose estimation approach has smaller pose error in this range of views, the algorithms denoted by A

(PETM-A, PASM-A, ASMM-A) demonstrate superior recognition rates than those in the other views. By the hybrid strategy, the algorithms denoted by B (PETM-B, PASM-B, ASMM-B) have approximately the same pose error and recognition rate in all orientations. And, because the tolerance of pose error is greater in cases close to lateral view, the quasi mirror morphing process which we design for side region can still work well in these cases. As shown in Fig. 15, the proposed algorithms (ASMM-A and ASMM-B) still have high recognition rates in the neighborhood of the lateral view. In ASMM-B, a recognition rate exceeding 95% can be achieved in all orientations, even in the cases close to the lateral view.

Figure 16 compares the recognition rates of ASMM-B (blue curve), PASM-B (green curve), and QRAFM with 45° and 135° templates (pink curve) over all orientations. According to the analysis of Fig. 14, the recognition rate of QRAFM decreases in conjunction with the increment of pose-difference between the template and the tested cars. For the QRAFM algorithm with the 45° template, the recognition rate is over 80% when the orientation of the tested car is within [40°, 50°]. If the orientation of the tested car falls in the range of $30° \pm 5°$ or $60° \pm 5°$, the recognition rate decreases to smaller than 70%. Moreover, the recognition rate is lower than 50% when the orientation of the tested car is smaller than 20° or larger than 70°. ASMM obviously outperforms QRAFM.

Since the QRAFM algorithm exhibits superior performance when the pose differences between the template and the tested cars are limited, we combine our proposed pose estimation approach with QRAFM and compare the integrated algorithm with ASMM. The integrated algorithm is termed PE-QRAFM (pose estimation-quasi rigid alignment and flexible matching). The recognition rate of PE-QRAFM which is shown as the dashed pink curve in Fig. 16 can be improved averaging to approximately 90%; however, our proposed algorithm ASMM achieves a superior recognition rate averaging around 98%. Therefore, even though the QRAFM algorithm is assisted by the proposed pose estimation approach, our proposed ASMM algorithm which utilizes the mirror morphing technique is superior to the QRAFM algorithm for car model recognition, especially in the cases of cars with similar shape and color.

### 6.3.4 Computation load

Our proposed system was implemented on an AMD Athlon II X2 240, 2.0 GB RAM, 2.81 GHz PC with Language C environment. The average computation time of each step performed on a tested image (image size $= 150 \times 500$ pixels) is reported in Table 3. The computation time for the whole system is about 0.530 s and such computation time is applicable to most real systems.

## 7 Conclusions

Recognizing car models remains a challenge nowadays, because cars in the same class tend to appear similar. In addition, the shape features of vehicles vary significantly from those of the template under different poses. Pose and center estimations can be performed to limit the influence of the pose variation. However, very accurate pose and center estimation is hardly to be obtained in practice and even a little deviation in the estimated pose and center would degrade the recognition performance significantly.

In this paper, we present a mirror morphing scheme which utilizes the symmetric property of cars to eliminate the pose variation and center bias. Therefore, the difficult pose problem can be accommodated. Mirror morphing requires knowing the locations of shape feature of cars first. We adopted ASM to acquire the shape information. An effective approach to obtain proper initial estimation of pose angle and center position is also proposed for ASM initialization. The estimated pose angle and center position provide a good start for ASM to lead to the right direction to find correct shape feature.

In experiments, our proposed pose estimation approach provides much smaller estimation error than the conventional pose estimation approach. Through the ASM processing, more accurate estimation of the center position can be obtained and the recognition performance of car models can be further improved. When the mirror morphing technique is added, the recognition rate can increase to reach a very high performance level. Our proposed system can achieve a very good recognition rate (>95%) with very low probability of false alarm over all orientations in the cases of the cars with similar shape and color.

The techniques which we present in this paper, including the initial pose and center estimation, shape feature extraction by ASM, and mirror morphing, can work together to form up an advanced system for car model recognition. Mirror morphing is the core function of our proposed system and it can also cooperate with other pose and center estimation or shape feature extraction algorithms to achieve even better performance. The experimental results show that the proposed pose and center estimation approach and the ASM technique which we adopted can support our mirror morphing process working effectively.

The proposed system is mainly designed for cars exhibiting symmetry in the front and rear regions. The system can recognize sedans, station wagons, vans, etc., well in all orientations. A number of vehicle types, such as SUVs, Jeeps, and trucks, have the symmetric properties in the front region, but usually lack symmetry in the rear region. Our proposed system is only able to recognize such vehicles in a limited number of orientations, in which the front region of the tested car is visible.

**Table 3** Average computation time of each step in the proposed system

| Step | (1) Pose & center estimation | (2) Shape feature extraction | (3) Car region segmentation | (4) Mirror morphing | (5) Template matching | Whole system |
|---|---|---|---|---|---|---|
| Computation time (s) | 0.072 | 0.140 | 0.026 | 0.283 | 0.009 | 0.530 |

Our proposed approach performs well for car images captured from the ground level or a slightly higher view. In many useful situations such as holding a camera in hands, mounting a camera on a tripod, or placing a camera on a driving car, the proposed approach achieves satisfactory recognition accuracy for car models. In surveillance systems, cameras are generally deployed at fixed locations with fixed heights, so the vertical view angle is limited to within a small range. The proposed approach can still improve the car model recognition rate. In more challenging situations where cameras are set up in locations with various vertical view angles, a more intelligent pose estimation algorithm should be developed and more templates covering a wide range of vertical view angles should be prepared. Cases involving a variety of vertical view angles will be dealt with in the future. Furthermore, our proposed approach is able to accommodate general lighting and weather variations. Recognizing cars under extremely various lighting and weather environments is another future work of this paper.

## References

1. Zheng, D., Zhao, Y., Wang, J.: An efficient method of license plate location. Pattern Recogn. Lett. **26**(15), 2431–2438 (2005)
2. Chang, S.L., Chen, L.S., Chung, Y.C., Chen, S.W.: Automatic license plate recognition. IEEE Trans. Intell. Transp. Syst. **5**(1), 42–53 (2004)
3. Anagnostopoulos, C.E., Anagnostopoulos, I.E., Psoroulas, I.D., Loumos, V., Kayafas, E.: License plate recognition from still images and video sequences: a survey. IEEE Trans. Intell. Transp. Syst. **9**(3), 377–391 (2008)
4. Hoffman, C., Dang, T., Stiller, C.: Vehicle detection fusing 2d visual features, pp. 280–285. IEEE Intelligent Vehicles Symposium (2004)
5. Betke, M., Haritaoglu, E., Davis, L.S.: Real-time multiple vehicle detection and tracking from a moving vehicle. Mach. Vis. Appl. **12**, 69–83 (2000)
6. Santos, D., Correia, P.L.: Car Recognition Based on Back Lights and Rear View Features, pp. 137–140. Workshop on Image Analysis for Multimedia Interactive Services (2009)
7. Wen, X., Zhao, H., Wang, N., Yuan, H.: A rear-vehicle detection system for static images based on monocular vision, pp. 1–4. International Conference Control, Automation, Robotics and Vision (2006)
8. Li, X.Z., Zhang, G.M., Fang, J., Wu, J.A., Cui, Z.M.: Vehicle color recognition using vector matching of template, pp. 189–193. International Symposium Electronic Commerce and Security (2010)
9. Kim, K.J., Park, S.M., Choi, Y.J.: Deciding the number of color histogram bins for vehicle color recognition, pp. 134–138. IEEE Conference Asia-Pacific Services Computing (2008)
10. Zhao, W., Chellappa, R., Rosenfeld, A., Phillpse, P.J.: Face Recognition: A Literature Survey. pp. 399–458. ACM Computing Surveys, New York (2003)
11. Chellappa, R., Sinha, P., Phillips, P.J.: Face recognition by computers and humans. IEEE J. Comput. Soc. **43**(2), 46–55 (2010)
12. Voit, M., Nickel, K., Stiefelhagen, R.: A bayesian approach for multi-view head pose estimation, pp. 31–34. International Conference Multi-sensor Fusion and Integration for Intelligent Systems (2006)
13. Voit, M., Nickel, K., Stiefelhagen, R.: Multi-view head pose estimation using neural networks, pp. 347–352. Canadian Conference on Computer and Robot Vision (2005)
14. Hou, T., Wang, S., Qin, H.: Vehicle matching and recognition under large variations of pose and illumination, pp. 24–29. International Conference Computer Vision and Pattern Recognition, June (2009)
15. Lai, A.H.S., Fung, G.S.K., Yung, N.H.C.: Vehicle type classification from visual based dimension estimation, pp. 201–206. IEEE International System Conference (2001)
16. Ozuysal, M., Lepetit, V., Fua, P.: Pose estimation for category specific multiview object localization, pp. 778–785. IEEE Conference on Computer Vision and Pattern Recognition (2009)
17. Arie-Nachimson, M., Basri, R.: Constructing implicit 3D shape models for pose estimation, pp. 1341–1348. International Conference on Computer Vision (2009)
18. Hutter, M., Brewer, N.: Matching 2-D ellipses to 3-D circles with application to vehicle pose identification, pp. 153–158. International Conference on Image and Vision Computing, New Zealand (2009)
19. Prokaj, J., Medioni, G.: 3D model based vehicle recognition, pp. 1–7. Workshop on Applications of Computer Vision (2009)
20. Guo, Y., Hsu, S., Sawhney, H.S., Kumar, R., Shan, Y.: Robust object matching for persistent tracking with heterogeneous features. IEEE Trans. Pattern Anal. Mach. Intell. **29**(5), 824–839 (2007)
21. Kuehnle, A.: Symmetry-based recognition of vehicle rears. Pattern Recogn. Lett. **12**(4), 249–258 (1991)
22. Zielke, T., Brauckmann, M., Seelen, W.V.: Intensity and Edge-based Symmetry Detection Applied to Car-following. Lecture Notes in Computer Science, vol. 588, pp. 865–873. Springer, Berlin (1992)
23. Kim, D.S., Chien, S.I.: Automatic car license plate extraction using modified generalized symmetry transform and image warping, vol. 3, pp. 2022–2027. IEEE International Symposium on Industrial Electronics (2001)
24. Kadow, U., Schneider, G., Vukotich, A.: Radar-vision based vehicle recognition with evolutionary optimized and boosted features, pp. 13–15. IEEE Intelligent Vehicles Symposium (2007)
25. Chang, H.Y., Fu, C.M., Huang, C.L.: Real-time vision-based preceding vehicle tracking and recognition, pp. 514–519. IEEE Intelligent Vehicles Symposium (2005)

26. Wan, S., Kodagoda, S., Sehestedt, S.: Multiple cue based vehicle detection and tracking for road safety, pp. 12–14. International Conference on Intelligent Technologies (2007)

27. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models—their training and application. Comput. Vis. Image Underst. **61**(1), 38–59 (1995)

28. Cootes, T.F., Page, G.J., Jackson, C.B., Taylor, C.J.: Statistical grey-level models for object location and identification. Image Vis. Comput. **14**(8), 533–540 (1996)

29. Liu, S.C., Fu, C.W., Chang, S.: Statistical change detection with moment under time-varying illumination. IEEE Trans. Image Process. **7**(9), 1258–1268 (1998)

30. Zhang, D.S., Lu, G.: Segmentation of moving objects in image sequence: a review. Circuits Syst. Signal Process. **20**(2), 143–183 (2001)

31. Chien, S.Y., Ma, S.Y., Chen, L.G.: Efficient moving object segmentation algorithm using background registration technique. IEEE Trans. Circuits Syst. Video Technol. **12**(7), 577–586 (2002)

32. Chen, B.S., Lei, Y.Q., Li, W.W.: A novel background model for real-time vehicle detection, vol. 2, pp. 1276–1279. IEEE International Conference Signal Processing (2004)

33. Leotta, M.J., Mundy, J.L.: Predicting high resolution image edges with a generic, adaptive, 3-D vehicle model, pp. 1311–1318. International Conference on Computer Vision and Pattern Recognition (2009)

34. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 681–685 (2001)

35. Wolberg, G.: Image morphing: a survey. Vis. Comput. **14**, 360–372 (1998)

36. Beier T., Neely S.: Feature-based image metamorphosis, pp. 35–42. SIGGRAPH (1992)

37. Zhao, S., Gao, W., Shan, S., Yin, B.: Enhance the alignment accuracy of active shape models using elastic graph matching, pp. 52–58. International Conference on Biometric Authentication (2004)

38. Zheng, Z.L., Yang, F.: Enhanced active shape model for facial feature localization. International Conference on Machine Learning and Cybernetics, vol. 5, pp. 2841–2845 (2008)

39. Gu, H.Z., Kao, Y.W., Lee, S.Y., Yuan, S.M.: HVPN: the combination of horizontal and vertical pose normalization for face recognition, pp. 367–378. International Conference on Multimedia Modeling (2009)

40. Brunelli, R.: Template Matching Techniques in Computer Vision: Theory and Practice. Wiley, New York (2009)

41. Dataset for car model recognition. http://www.cs.nctu.edu.tw/~hcku/

## Author Biographies

**Hui-Zhen Gu** received her MS degree in Computer Science and Information Engineering from National Chiao Tung University, Taiwan, in 2007. She is currently a Ph.D. candidate of Computer Science and Information Engineering in National Chiao Tung University, Taiwan. Her research interests include computer vision, content-based video retrieval, multimedia information system, and intelligent surveillance system.

**Suh-Yin Lee** is a Professor at Computer Science and Information Engineering in National Chiao Tung University, Taiwan. She received her M.S. degree in Computer Science from University of Washington, Seattle, U.S.A., in 1975, and her Ph.D. degree in Computer Science from Institute of Electronics, National Chiao Tung University, Taiwan. Her research interests include content-based indexing and retrieval, distributed multimedia information system, mobile computing, and data mining.