IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 22 February 2023, accepted 24 March 2023, date of publication 29 March 2023, date of current version 13 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3262803

RESEARCH ARTICLE

Template-Based 2D High-Precision Model Reconstruction of Car Body

BO WANG[1], ZHIYING YANG[1], FRANCA GIANNINI[2], MARINA MONTI[2], AND BAOJUN LI[3], (Member, IEEE)

[1]School of Automobile and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430065, China
[2]Institute for Applied Mathematics and Information Technologies, National Research Council, 16149 Genoa, Italy
[3]School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China

Corresponding author: Bo Wang (wangbosofia@gmail.com)

This work was supported by the National Natural Science Foundation of China under Grant 51905389.

ABSTRACT Car body design includes the construction of car sketches, 2D models and 3D models. The establishment of high-quality car body models usually relies on CAD software, which requires the proficient operating skills, and the efficiency is relatively low compared with automatic modeling. The time cost of communication between the designer and the model engineer will also affect the modeling efficiency. How to quickly and automatically generate high quality parametric models from car sketches or images efficiently has become a development direction for the automotive design. In this paper, we propose a template-based reconstruction method of 2D high-precision car body model. On the basis of the established 2D model database, a coarse model is achieved by the improved Orthogonal Matching Pursuit algorithm with few key points which are obtained by deep neural network; further, according to the proposed auto-fitting optimization algorithm of cubic Bezier curve, the modeling process of "from coarse to fine" is realized combining edge information. The proposed template-based 2D high-precision model reconstruction algorithm of the car body can greatly reduce the modeling time under the given modeling accuracy.

INDEX TERMS Bézier curve fitting, key points extraction, orthogonal matching pursuit algorithm, car body reconstruction.

I. INTRODUCTION

When designing a new car, designers often refer to existing models, and lots of image resources can help designers find corresponding inspiration. When designing a new car, the abstract basic curves will be started firstly. These curves can convey the basic feature information of the car body, such as the outer contour line of the car body, the lines of the windows and doors, etc., and are also the perception lines that significantly affect the perception of the product [1]. Nowadays, the automobile industry generally adopts computer-aided technology for product design and manufacture, which shortens the development cycle and greatly improves the product quality. The 2D parametric model of the car body, as the 2D representation of the car body shape, is the basis of the 3D reconstruction and plays a vital role in the automobile design process. For the wide variety of car body shapes nowadays, the 2D parametric model of the body is also useful for analyzing the similarities and differences between different models, and facilitating the diversity of designs [2].

The mainstream car body design usually uses CAD software. This method shortens the development cycle compared to traditional hand-drawn image, but it requires technicians have a certain knowledge of modeling and be proficient in CAD software operation. Moreover, moving from design sketch to model construction involves different staff, and communication and iterative modification of design sketches can consume a lot of time. It is very difficult for a person with no experience in CAD software operation and modeling to design his own car based on 2D pictures, and these problems become a major obstacle to include end user in the car design. In order to solve the technical barrier of user independent design, this paper attempts to generate high-quality 2D parametric model from a single car body image. The 2D model

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wang.

obtained in this paper can also provide rich data support for subsequent 3D modeling.

This paper introduces a fully automatic method for extracting a 2D parametric model of a car body. The method combines the edge information of the car image and the knowledge reuse of the template database to extract the 2D parametric model of the car body. The whole extraction process follows the idea of ''from coarse to fine'': 1) few key points which are obtained by deep neural network as the input; 2) the coarse extraction of the model based on the Orthogonal Matching Pursuit (OMP) algorithm; 3) the optimized and refined extraction model by our Bézier curve auto-fitting algorithm combining edge information from the image. The complete workflow applied for the model reconstruction is shown in Fig. 1.

II. RELATED WORKS

In this section, we firstly review the rough model generation method by compressed sensing theory, followed by related work in the areas of key points detection, B-spline curve fitting, edge detection, and finally the fine 2D model reconstruction.

A. COMPRESSED SENSING THEORY

The OMP algorithm is selected to calculate the initial car body model. Compressed sensing (CS) theory is an emerging signal compressive sampling technique proposed by Donoho [3] in 2006. This theory solves the problem of signal reconstruction on the premise of signal sparsity or compressibility. As a class of CS reconstruction algorithms, greedy pursuit algorithm has fast convergence speed and high reconstruction accuracy. The greedy pursuit algorithm is to update the support set through greedy iteration and gradually approximate the original solution, such as the gradient pursuit (GP) algorithm [4], the matching pursuit (MP) algorithm [5], and the OMP algorithm [6], stagewise orthogonal matching pursuit (StOMP) algorithm [7] and regularized orthogonal matching pursuit (ROMP) algorithm [8] and so on.

B. KEY POINTS DETECTION

The coarse extraction of the model needs to use the key points of the body as input, so the extraction for the key points is also very important in this paper. The key point extraction method in this paper refers to the facial landmark detection. Facial landmark detection algorithms are divided into traditional detection algorithms and deep learning methods. Active Shape Model (ASM) is a classical facial landmark detection algorithm proposed by Cootes et al. [9], which abstracts the target object by a shape model. The ASM algorithm requires a manual calibration method to calibrate the training set. A shape model is obtained after training, and then the object-specific matching can be achieved by key points. By considering boundary information as the geometric structure of human faces, Wu et al. [10] presented a boundary-aware face alignment algorithm (LAB) to improve the detection accuracy. LAB derives the face landmarks from the boundary lines, which can largely avoid the ambiguity of landmark definitions. PFLD algorithm [11] investigates a neat model that has good detection accuracy in wild environments (e.g., unconstrained pose, expression, illumination, and occlusion conditions) and ultra-real-time speed on mobile devices.

C. BÉZIER CURVE FITTING METHOD

As the NURBS curve defined by several control points is widely used in industry design, the special form Bézier curve is selected as the basic element of the 2D parametric model of the car body in this paper. The cubic Bézier curve, which is the basic element of the body wireframe model in this paper, needs to be automatically generated based on the body parameter template combined with edge information of the image. At present, most of the fitting of B-spline curve is in the form of calculating B-spline curve under the premise of given data points. Such as the Progressive Iterative Approximation (PIA), which is a data fitting method with obvious geometrical significance, simple programming and stable convergence, and starts with the uniform cubic B-spline curve with break-even correction proposed by Qi et al. [12]. In the classical PIA method, the number of control vertices is equal to the number of data points to be fitted, but it is not suitable for fitting large-scale data points. Considering this problem, Lin and Zhang [13] proposed a generalized PIA method, which has the advantages of local nature and parallel computing, and is suitable for large-scale data point fitting. The Least Squares-based Progressive Iterative Approximation (LSPIA) method proposed by Deng and Lin [14] obtains a series of fitting curves (surfaces) by continuously adjusting the control vertices, based on the least squares fitting method: this method can efficiently and stably fit large-scale data points and significantly reduce the amount of calculation.

D. EDGE DETECTION METHOD

Extracting the 2D model of the car body from the image requires the edge information. In recent years, edge detection is crucial in various machine vision tasks, such as image segmentation [15], target detection [16], and target recognition [17]. Highly accurate edges also contribute to the construction of 2D models of the car body. Most edge detection algorithms can be divided into three categories: traditional edge operators, learning-based methods, and recently popular deep learning methods. Traditional edge detection operators [18], [19], [20], [21] detect edges by finding sudden changes in intensity, color, texture, etc. Learning-based methods utilize supervised models and handcrafted features to identify edges. For example, Dollár and Zitnick [22] proposed structured edges that can jointly learn the clustering of groundtruth edges and the mapping of image patches to cluster labels. Deep learning based methods use Convolutional Neural Network (CNN) to extract multi-level features. Such as the network designed by Liu et al. [23] makes full use of

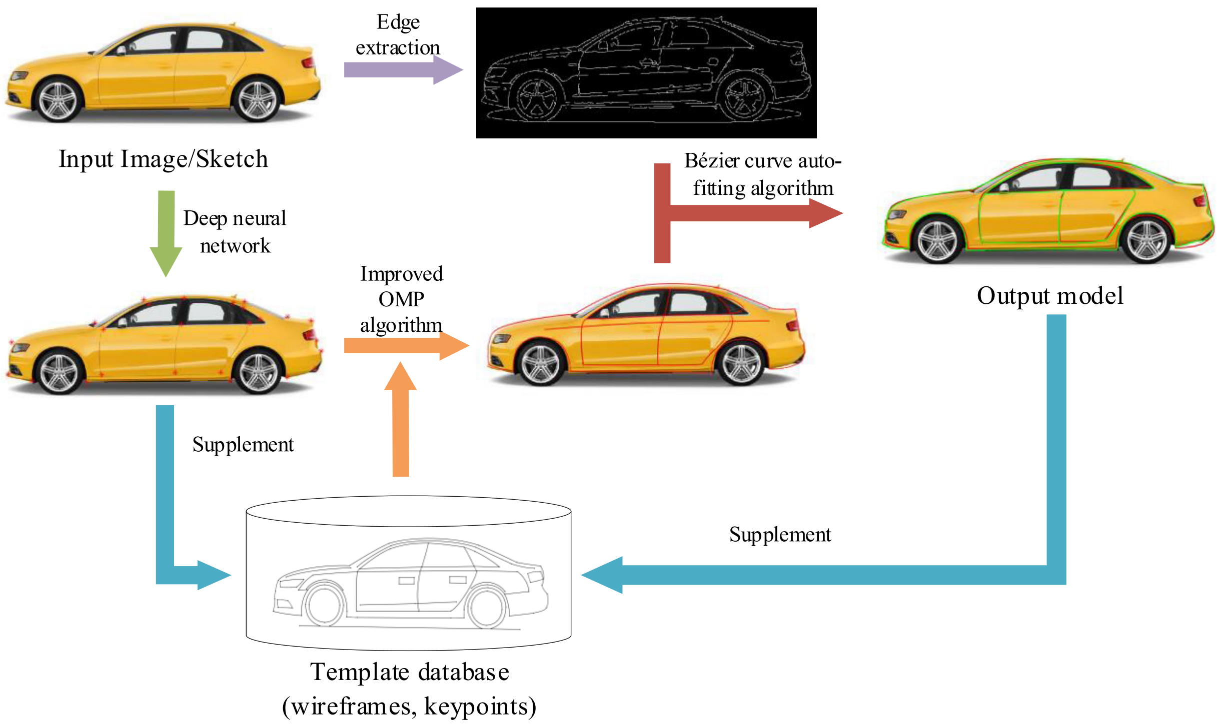

FIGURE 1. **Model reconstruction flow. The input image is downloaded from Google Plus, and sketch image can be rendered by a car image.**

the multi-scale and multi-level information of the target to perform inter-image prediction by combining all meaningful convolutional features holistically.

E. AUTOMOTIVE MODEL EXTRACTION

At the same time, many researchers have also carried out work on 2D model extraction. To explore the use of deformable car models, Smith et al. [24] created a series of 2D and 3D databases suitable for standardized templates containing separate meshes for the body, roof, and wheels, etc. Li et al. [25] proposed a method for extracting a parametric model of car feature lines from car side-view images, which can effectively recover the car styling features represented by cubic Bézier curves, but the process is not fully automatic as it still requires the user to spend lots of time to adjust some control points in the post-processing steps. Bluntzer et al. [26] proposed a method to identify, extract and interpret car feature lines from sketches for early aesthetic exploration of car style, which highlights some geometric differences between the character lines of French and German cars. Wang et al. [27] illustrates a method to automatically derive a 2D parametric model of the main characteristic lines of a car from images, blueprints or hand-made sketches of its side view, but it still needs minutes to obtain the model, and the complex algorithm is not conducive to reuse for other products. Wang et al. [28] proposes a curve-based image warping method to reshape the content of images, the paper also shows that the semantic editing of car images can be easily achieved by combining this method with the automatic identification of semantically important character lines.

This paper firstly extracts few specific key points from coarse to fine with deep learning method, then obtains an initial coarse extraction model through the improved OMP algorithm, finally, based on the quick Bézier curve auto-fitting algorithm for precise adjustment combining with the edge information from car image, a high-precision 2D wire-frame model of the car body is finally generated quickly, and which can be easily extended to other products.

The method proposed in this paper can achieve the reconstruction of a corresponding 2D car body model from a single car body side view, which can significantly reduce modeling time and improve modeling efficiency compared to CAD modeling.

III. MODEL DATABASE AND TEMPLATE

2D model database in [25] has been still selected in this paper, which includes several vehicle templates of different brands and models. Each template referred to the database consists of 97 cubic Bézier curves which contains 4 control points shown in Fig. 2, including the outer contour structure of the body, the internal feature structure and the detailed feature structure. Therefore the model contains 388 control points in total ignoring the coinciding end points of discrete two curves.

According to the parameterized templates and the improved OMP algorithm, the coarse model can be generated by few specific key points. Analyzing the structure of car

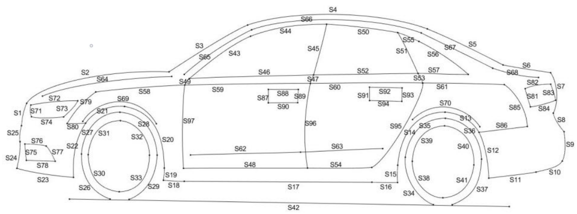

FIGURE 2. **2D model of the car body.**

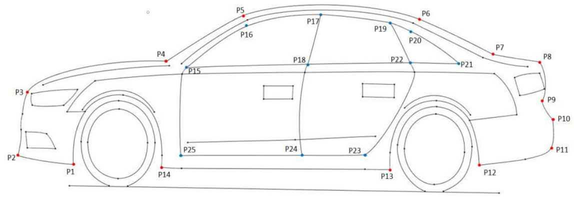

FIGURE 3. **The key points definition method in [25].**

body side view, the specific key points are determined based on the 2D model. Further, template database should be calibrated firstly for the extraction of key points by PFLD deep learning and the coarse model generated by the improved OMP algorithm respectively.

A. DEFINITION OF KEY POINTS FOR BODY SIDE VIEW

This paper mainly focuses on the modeling of the main features of the car body, including the main feature lines such as the outer contour of the car body, windows and doors, etc.. The interior and detailed feature structures such as lights have not been considered in detail. Therefore, in order to ensure the robustness of the model reconstruction algorithm by counting the reconstruction error of the 2D feature line, the 25 key points are selected according to the definition in [25], which is shown in Fig. 3.

B. TEMPLATE DATABASE CALIBRATION

The template database used in this article only includes sedan and SUV. If you want to try other types of vehicle models (such as trucks) in the future, transfer learning can be used to redefine the model.

Since the template database is not established with a unified reference point, which is not conducive to the model reconstruction, the entire database needs to be calibrated firstly. Considering the common structure of all models and the efficiency of calculation, the front wheel center PF is set as the coordinate origin, and the model is then slightly rotated to set the line between PF and the rear wheel center PR parallel to the x axis.

The wheels in the template are spliced by 8 cubic Bézier curves as shown in Fig. 4. For example, S26 – S29 in Fig. 2 are the front wheel lines, and S30–S33 in Fig. 2 are the front wheel hub lines. With the front wheel center $PF(pf_x, pf_y)$ and the rear wheel center $PR(pr_x, pr_y)$, the model can be translated and rotated to our calibration type. Computing

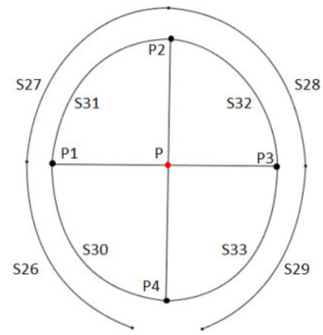

FIGURE 4. **The composition of the wheel and the determination of the wheel center.**

wheelbase as $L = |PF - PR|$, translation distance as $-(pf_x, -pf_y)$, and the cosine and sine value of the rotation angle as $\frac{pr_x - pf_x}{L}$ and $\frac{pr_y - pf_y}{L}$, then any control point $p(p_x, p_y)$ of the model is translated to $newp(newp_x, newp_y)$, with $newp_x = \frac{(p_x - pf_x) * (pr_x - pf_x)}{L} + \frac{(p_y - pf_y) * (pr_y - pf_y)}{L}$, $newp_y = \frac{(p_y - pf_y) * (pr_x - pf_x)}{L} + \frac{(p_x - pf_x) * (pr_y - pf_y)}{L}$. When performing rotation correction, the rotation should be performed at a small angle to avoid model inversion.

IV. COARSE MODEL GENERATION METHOD

Manual extraction of key points requires operators with extensive experience in point selection and a large time cost. Therefore, this paper considers to use deep neural network to automatically generate feature points. For automatically reconstruct a 2D high-precision model of car body only by inputting the body image, firstly the initial key points are estimated by MobileNet-v2, then the coarse model is generated by the improved OMP algorithm.

A. COARSE TO FINE KEY POINTS EXTRACTION BY MobileNet-v2

Considering that the modeling method proposed in this paper may be applied to mobile or embedded devices in the future, the neural network model cannot be too large, so it is crucial to study small and efficient neural network models. The current research on small models is divided into two directions: 1) the trained complex models are compressed to obtain small models; 2) small models are designed directly. In any case, the goal is to reduce the model size while maintaining the model performance and improving the model speed.

In this paper, we refer to the network model in the PFLD algorithm [11], which uses Mobilenet-v2 as the backbone part of the network structure, increasing the expressiveness by fusing features from three different scales. Our training set contains 4000 car body images and the test set contains 200 car body images.

Because of the limitation in power computation, the images are compressed to 224 × 224 firstly, and the key points can be extracted under such size, then which will be amplified to the original size to obtain the coarse key points. However, the compressing and amplifying process reduce the extraction precision, with the lost of some feature information with the compressed pixel blocks, and further larger position error with the amplifying process.

In order to solve the above problem, this paper proposes a point-by-point extraction method. For every key point, cropping 224 × 224 pixels around the key point in the original image respectively, 25 separate deep models are computed with the training set, finally the cropping region for the test image is obtained according to the coarse position. This idea of "from coarse to fine" lifts the extraction accuracy for all the key points with tackling the original images.

With the better extraction key points, the coarse model can be generated by sparse reconstruction algorithm below.

B. IMPROVED OMP ALGORITHM

OMP algorithm is one of the classical algorithms in the field of compressive sensing, which is mainly used to solve the sparse reconstruction problem of signals. In this paper, we use this algorithm to reconstruct the coarse model by sparse key points.

The traditional OMP algorithm is based on the greedy iterative method to determine the columns of the sensing matrix, so as to ensure that the columns selected in the later stage are as close as possible to the redundant vector at the current stage. The redundant part in the selected vector is removed under enforced multiple iterations, ensuring that the process continues until the number of iterations and sparsity K are the same before stopping.

If it is a signal processing problem, the calculation result of OMP algorithm is the reconstructed signal. However, in this paper, the OMP algorithm will get a sparse representation coefficient θ at the end of the iteration. The sparse representation coefficient θ is defined as the contribution of the model, and which is the functional relationship to be computed. Among them, θ contains K non-zero elements, which indicates that K models are involved in the calculation At the same time, since the solved θ is sparse, this paper also adds the full contribution of the templates to the algorithm, which makes up for the problem that the traditional OMP algorithm cannot satisfy all models to participate in the calculation.

Firstly, the geometric information of a single model is converted into a shape vector $s_i = (x_1, y_1, x_2, y_2, \ldots, x_l, y_l)^T$, which contains the coordinate values of all control points of the model. Thus the entire database can be transformed into a shape matrix $S = (s_1, s_2, \ldots, s_{215})^T$, which contains templates for 215 car models. Secondly, the key point information of the input image is expanded into a key point vector $z_m = (x_1, y_1, x_2, y_2, \ldots, x_m, y_m)^T$, which represents the coordinate values of a few of input key points. From the shape matrix S, a sub-matrix Z containing m body key points in each model can be extracted, and the functional relationship f between z_m and Z can be established by OMP algorithm. This functional relationship can also be used for s_i and S, the coarse extraction of the model can then be obtained.

According to the pseudocode of Algorithm 1, it is clear that the time complexity of the improved OMP algorithm is O(n). Due to the need to manually define the value of K in the algorithm, different K values can lead to different coarse extraction models. The bottleneck of how to achieve the best results with simultaneous adaptive K needs to be addressed in future research.

Algorithm 1 Improved OMP Algorithm

Input: The submatrix of car body key points (Z), The key-point vector (z_m), The sparsity (K);
Output: Contribution of the model (θ) ;

Initializing residuals $r_0 = z_m$, index set $A_0 = \emptyset$, number of iterations $t = 1$;
if $K = 0$ **then**
  $\emptyset = Z^{-1} \times z_m$;
end if
if $K < 0$ **then**
  return false;
end if
if $K > 0$ **then**
  while $t \leq K$ **do**
    Finding the index λ corresponding to the maximum value of the inner product of the residual r and each column Z_i of the sensing matrix, $\lambda_t = argmax_{i=1,\ldots,n} |\langle r_{t-1}, Z_i \rangle|$;
    Updating index set and submatrix, $A_t = A_{t-1} \cup \{\lambda_t\}$, $Z_t = Z_{t-1} \cup \{Z_\lambda\}$;
    Finding the least squares solution to $z_m = Z_t * \theta_t$, $\theta_t = argmin\|z_m - Z_t * \theta_t\| = (Z_t^T Z_t)^{-1} Z_t^T z_m$;
    Updating residuals r_t, $r_t = z_m - Z_t * \theta_t = z_m - Z_t (Z_t^T Z_t)^{-1} Z_t^T z_m$;
    $t = t + 1$;
  end while
end if

V. FINE MODEL GENERATION METHOD

After completing the coarse extraction of the model, the model needs to be optimized. In order to extract the 2D wire-frame model with Bezier curve as its basic element from the edge information of car image, this paper develops the Bézier curve auto-fitting algorithm. The mathematical expression of cubic Bézier is shown in Eq. 1.

$$\begin{aligned} B(t) = (1-t)^3 \times P0 + 3t(1-t)^2 \times P1 \\ + 3(1-t)t^2 \times P2 + t^3 \times P3, t \in [0, 1] \end{aligned} \tag{1}$$

where t is the auxiliary parameter; $P0$ is the starting control point; $P1$ and $P2$ are the intermediate control points; $P3$ is the ending control point.

A. CALCULATION OF BIAS MATRIX AND MEAN FITTING LENGTH

In order to adjust the control points through an iterative method, we firstly calculate the length and directions for all points in the area to the nearest edge point, and recorded as a bias matrix, then we minimize the mean fitting length of the objective curve in the iterative progress.

The area where the real line is located as $I(M, N)$, where M and N represent the length and width of the area (if (i, j) is an edge point, then $I(i, j) = 1$; otherwise $I(i, j) = 0$). The bias matrix noted as $BM(4, M, N)$, for any point (i, j) in the area, $BM(4, i, j)$ is calculated by the length to the nearest line point in the 4 directions of $\pm x$ and $\pm y$. Considering the proportion of edge points is relatively small, the specific calculation process is as follows. Initializing $BM(4, M, N) = \infty$, for any edge point (i, j). If $I(i-ki, j) == 1\&\&I(i-k, j) == 0(\forall k < ki)$, $BM(1, i-k, j) = k(k = 1, 2, \ldots, ki)$; if $I(i+ki, j) == 1\&\&I(i+k, j) == 0(\forall k < ki)$, $BM(2, i+k, j) = k(k = 1, 2, \ldots, ki)$; if $I(i, j-ki) == 1\&\&I(i, j-k) == 0(\forall k < ki)$, $BM(3, i, j-k) = k(k = 1, 2, \ldots, ki)$; if $I(i, j+ki) == 1\&\&I(i, j+k) == 0(\forall k < ki)$, $BM(4, i, j+k) = k(k = 1, 2, \ldots, ki)$.

Discretizing a given initial cubic Bézier curve to n uniform discrete points $\{P_1, P_2, \ldots, P_n\}$, the corresponding curve bias matrix $CBM(:, \{Pi_x\}, \{Pi_y\})$ is calculated directly from BM, and the MeanFittingLength (abbreviated as MFL) of the curve is defined as Eq. 2.

$$MFL = sum(min(CBM))/n \tag{2}$$

In Eq.2, the numerator represents the sum of the bias length of the discrete points of the curve. The smaller the bias length, the smaller the length to the actual edge. Therefore, the smaller the average fitting length, the closer fitting curve to the ideal edge.

B. CALCULATION OF THE CONTROL POINT UPDATING INCREMENTS AND UPDATE THE CONTROL POINTS

In order to make the final curve more accurate, the piecewise calculation idea is adopted to adjust the control points. Let the objective fitting cubic Bezier curve be $P_u = c(u)$, with the four control points: $P0$, $P1$, $P2$ and $P3$. let $L1 = |P1 - P0|$, $L2 = |P2 - P1|$, $L3 = |P3 - P2|$, then the piecewise parameters $u1 = L1/(L1+L2+L3)$, $u2 = (L1+L2)/(L1+L2+L3)$. when updating $P0$, the range of u in the curve expression is $(0 : u1)$; when updating $P1$, the range of u is $(u1/2 : u2)$; when updating $P2$, the range of u is $(u1 : (u2+1)/2)$; when updating $P3$, the range of u is $(u2 : 1)$; $oldP$ is the control point to be updated and the updating increments are Δx and Δy respectively, which are computed iteratively according to the fitness in the four directions.

(1) Firstly considering the increment in the direction of x, $Lvalue$ and $Rvalue$ are respectively the increments of $-x$ and $+x$ direction. Let $MI_1 = \{q|CBM(1, q) \leq MaxError\}$ and $MI_2 = \{q|CBM(2, q) \leq MaxError\}$ be the set of discrete points satisfying the minimum bias constraint in the $-x$ and $+x$ direction respectively. The corresponding increments $Lvalue$ and $Rvalue$ are calculated as Eq.3 and Eq.4.

$$Lvalue = \frac{sum(CBM(1, MI_1))}{|MI_1|} \tag{3}$$

$$Rvalue = \frac{sum(CBM(2, MI_2))}{|MI_2|} \tag{4}$$

TABLE 1. Time and error of Bézier curve auto-fitting algorithm.

Number of iterations	Time(ms)	Error(MSE)
1	1.476	24.9447
2	2.934	1.3558
5	6.296	0.4194

where$|MI_1|$ is the number of elements of the set vector MI_1 ; $|MI_2|$ is the number of elements of the set vector MI_2.

If $Lvalue < Rvalue$, the average bias length in the $+x$ direction is small therefore the fitting of the $+x$ direction is better, which means the point needs to be updated to the right and the update increment $\Delta x = Lvalue$. Else if $Lvalue > Rvalue$, the update increment $\Delta x = -Rvalue$. Else if $Lvalue = Rvalue$, the update increment $\Delta x = 0$.

(2) Similarly, the increment of Δy can be calculated. Then the control point is updated to

$$newP = oldP + [\Delta x, \Delta y] \tag{5}$$

Calculating $newMFL$ of the updating curve with the new control points, if $newMFL > MFL$, then $newP = oldP$ and $MFL = newMFL$.

(3) The optimal control point location is calculated iteratively by steps 1) and 2) with the maximum number of iterations set to 5.

(4) The control point $P1$ is adjusted firstly, then the control point $P2$. Then repeating this process for the more accurate results.

Taking the adjustment of $P1$ as an example, the Bézier curve auto-fitting algorithm is shown in Algorithm 2.

As can be seen from the pseudocode of Algorithm 2, the time complexity of the algorithm is O (n). The Bézier curve auto-fitting algorithm requires priori knowledge of coarse extraction models for model optimization, which is where the algorithm needs to be improved.

To verify the effectiveness of the algorithm, this method uses a given cubic Bézier curve for fitting experiments. The control points of the real Bézier curve in the experiment are $P0 = [5, 5]$, $P1 = [20, 50]$, $P2 = [80, 40]$, $P3 = [100, 5]$. And the control points of the given initial lines are $P0' = [5, 5]$, $P1' = [35, 0]$, $P2' = [65, 0]$, $P3' = [100, 5]$. Two interference lines are also given. The fitting results are shown in Fig. 5.

From the experimental results in Fig. 5, it can be seen that the real Bézier curve can be basically fitted after few iterations. The running time and fitting error of the algorithm are shown in Table 1, and after several cycles of experiments, it can be seen that each iteration takes 1.3ms on average.

C. MODEL ACCURATE ADJUSTMENT BASED ON Bézier CURVE AUTO-FITTING ALGORITHM

The line points of the car image are extracted by the edge detection algorithm in [22], as shown in Fig. 6b.

The relatively dense edges often affect the fitting accuracy with the bias length in the four directions may meet the conditions of the fitting error. Therefore, a chain-based approach

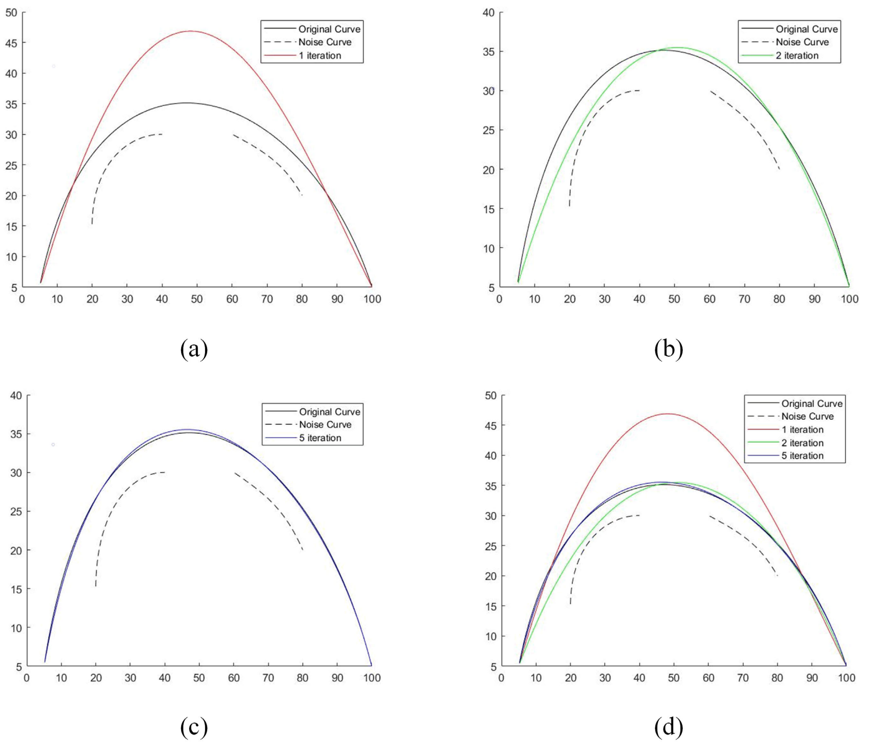

FIGURE 5. Cubic Bezier curve fitting process. (a) Curve fitting result after 1 iteration. (b) Curve fitting result after 2 iterations. (c) Curve fitting result after 5 iterations. (d) Summary of all fitting results.

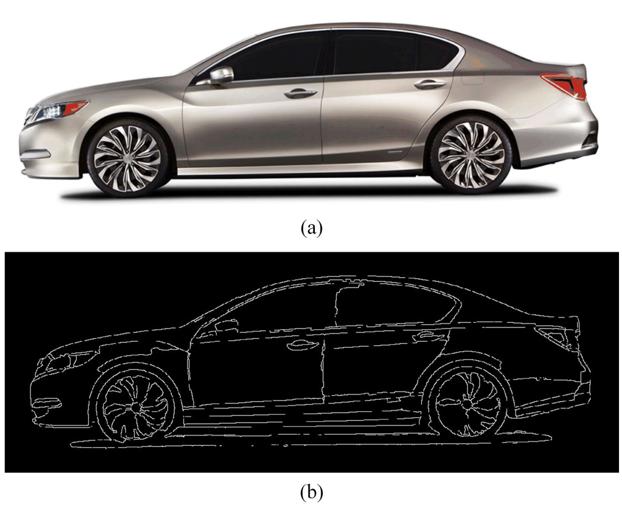

FIGURE 6. Edge information of a car body image. (a) Original image. (b) Edge information.

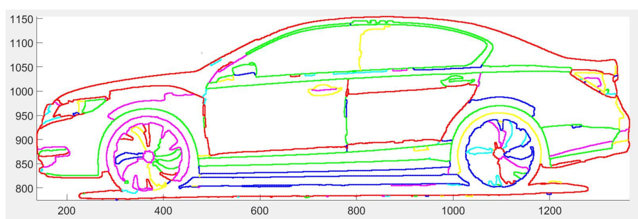

FIGURE 7. Chained body edge information.

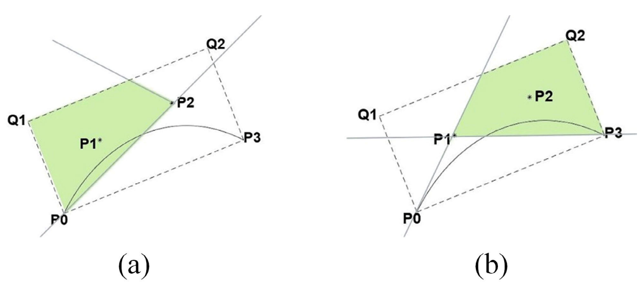

FIGURE 8. Intermediate control point adjustment area. (a) Adjustment area for $P1$. (b) Adjustment area for $P2$.

is adopted here to effectively solve this problem. In order to reduce the influence of noise points, the edge is chained by point chain connection using the method shown in [27], and the unimportant edge points in the image are further eliminated, as shown in Fig. 7. On the basis of chaining, the edge closest to the fitted curve can be locked from the beginning, thereby eliminating wrong fitting caused by the denseness of some individual edges. When calculating the bias matrix BM, the serial number of the corresponding chain is also recorded, and the corresponding intensity value is also saved in the chain serial number matrix C_L.

Algorithm 2 Bézier Curve Auto-Fitting Algorithm

Input: The initial Bézier curve, The four control points ($P0$, $P1$, $P2$ and $P3$);
Output: The new control points ($newP0$, $newP1$, $newP2$ and $newP3$) ;

Discretizing a given initial cubic Bézier curve to n uniform discrete points $\{P_1, P_2, \ldots, P_n\}$;
while $t \leq 5$ **do**
    Calculating the current average fitting length MFL of the curve according to Eq. 2
    Calculating the increment of Δx in the direction of x, where $Lvalue = \frac{sum(CBM(1,MI_1))}{|MI_1|}$, $Rvalue = \frac{sum(CBM(2,MI_2))}{|MI_2|}$
    if $Lvalue = Rvalue$ **then**
        $\Delta x = 0$
    end if
    if $Lvalue < Rvalue$ **then**
        $\Delta x = Lvalue$
    end if
    if $Lvalue > Rvalue$ **then**
        $\Delta x = -Rvalue$
    end if
    Similarly, calculating Δy, $newP = oldP + [\Delta x, \Delta y]$
    Calculating the current average fitting length $newMFL$ of the curve formed by the new control points $[P0, newP1, P2, P3]$
    if $newMFL > MFL$ **then**
        $newP = oldP$, $MFL = newMFL$
    end if
    $t = t + 1$
end while

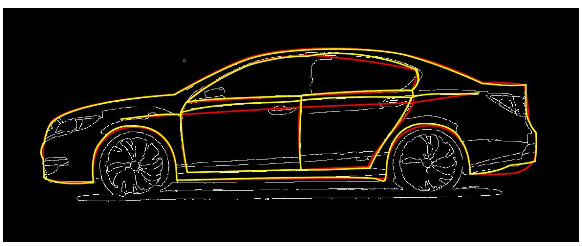

FIGURE 9. Accurately extracted body model. The red line is the coarse extracted model, and the yellow line is the optimized accurate model.

FIGURE 10. Parts with large coarse extracted errors can be better approximated to the real edge after optimization.

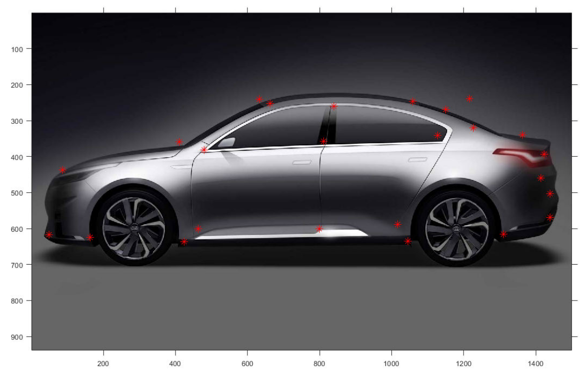

FIGURE 11. The coarse extraction of key points.

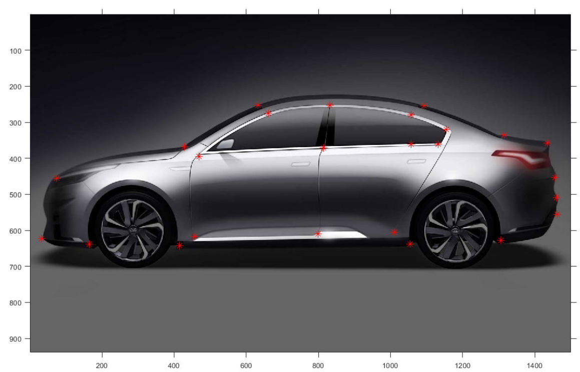

FIGURE 12. The fine extraction of key points.

The Bézier curve auto-fitting algorithm is applied to each feature line based on the coarse extraction model. So that the cubic Bézier curves in the car body are not distorted and too convex, the adjustment region is bounded in the calculation process. The constrained deformation region is shown in Fig. 8, where $[P0, Q1, Q2, P3]$ is the constrained rectangle. When adjusting $P1$, the adjustment region of $P1$ should fall in the green region in Fig. 8(a), and when adjusting $P2$, the adjustment region of $P2$ should fall in the green region in Fig. 8(b).

Further, the Bézier curve auto-fitting algorithm can also be used for the endpoints associated with 2, 3 or 4 lines. In a few seconds, a model that is closer to the real edge can be obtained imposing the fitting algorithm to the objective feature lines. The model result is shown in Fig. 9, and in Fig. 10 the error in the rough extraction of the model is relatively large. The yellow lines are the more accurate model after the fitting step and the red lines are the coarse results from improved OMP algorithm; therefore, the further fitting algorithm can well approximate the real edge.

VI. EXPERIMENTS

The experiments are performed on a PC with an Intel i7-11700 2.50 GHz CPU and NVIDIA GeForce RTX3090 GPU, and the algorithm compilation environment is Python and MATLAB.

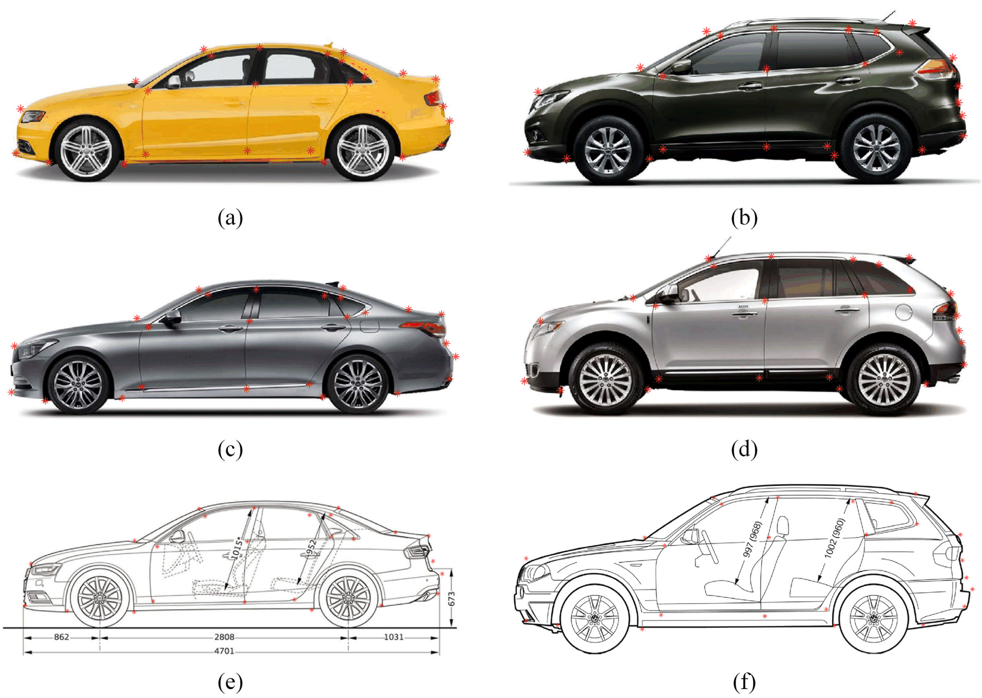

FIGURE 13. The key points extraction results. (a) The extraction result of sedan A. (b) The extraction result of SUV A. (c) The extraction result of sedan B. (d) The extraction result of SUV B. (e) The extraction result of sketch A. (f) The extraction result of sketch B.

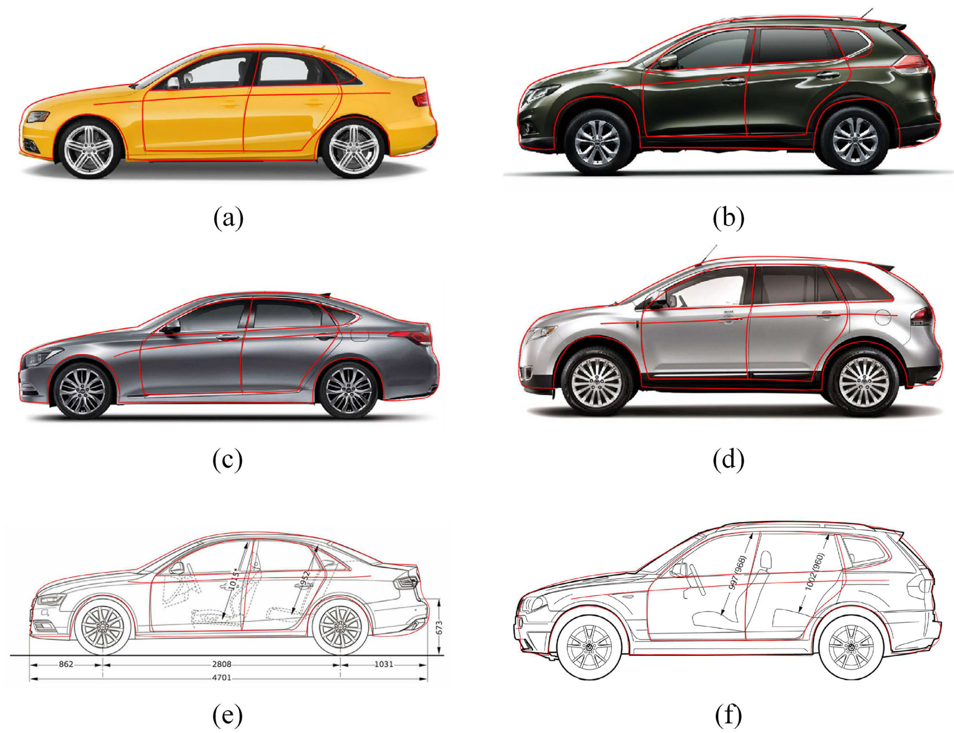

FIGURE 14. The coarse model extraction results corresponding to the images in Fig. 13.

A. KEY POINTS EXTRACTION BASED ON MobileNet-v2

The key points are extracted in two steps, coarse extraction to roughly identify the region, and fine extraction at every corresponding region. In the coarse step, the data set is compressed into 224 $\times$ 224 images, 25 coarse key points are extracted, and the result of restoring to the original image is shown in Fig. 11. In the fine step, 224 $\times$ 224 pixels region is cropped from the original image based on the 25 coarse key points

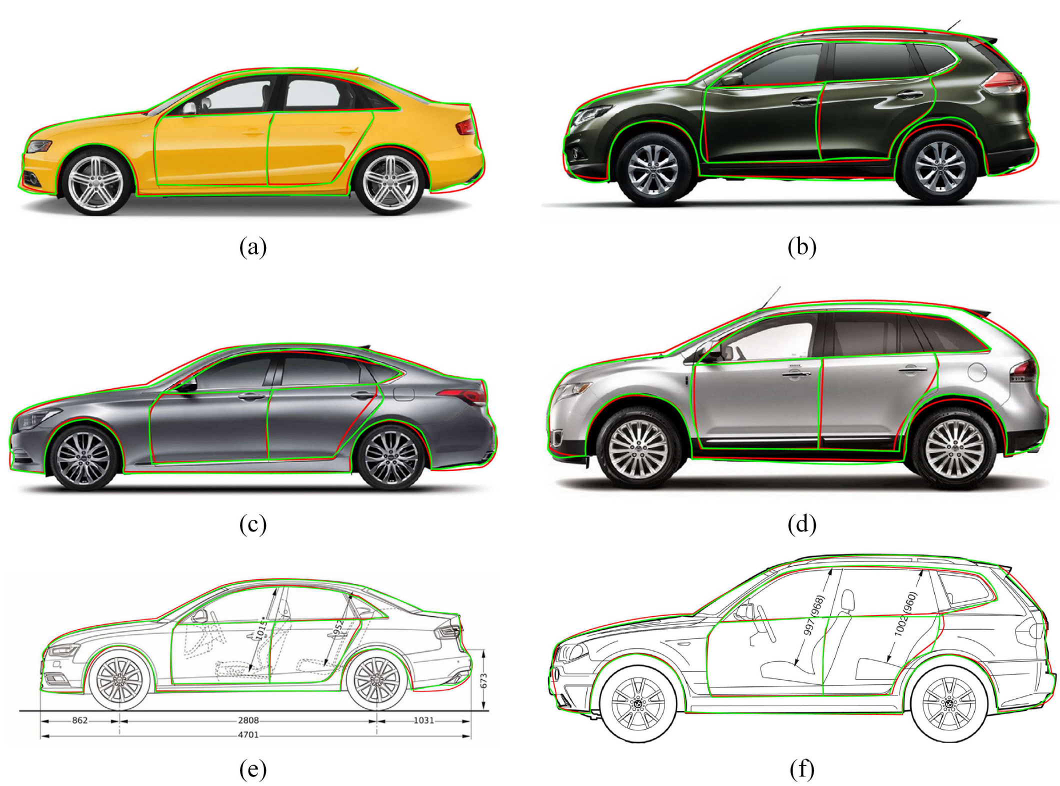

FIGURE 15. The fine model results corresponding to the images in Fig. 13.

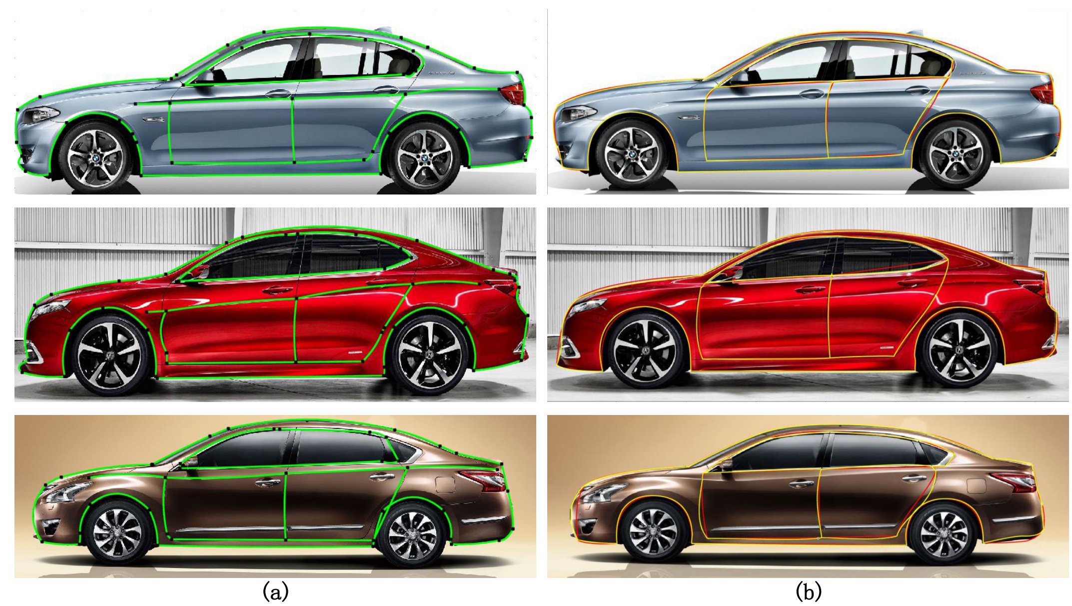

FIGURE 16. Comparison results with the method in [27]. (a) The results with the method in [27]. (b)The results with the proposed method.

positions, and the extracted better positions result is shown in Fig. 12 integrating 25 trained model.

The extraction errors of 200 images in the test set are counted. The data show that this ‘‘from coarse to fine’’ extraction method can control the error of key points within 10 pixels, as shown in Table 2.

In order to verify the applicability of the algorithm to different models, we used three sets of different images for

TABLE 2. The extraction errors of 200 images in the test set.

Error range(Number of pixels)	Proportion
0∼2	20%
2∼4	30%
4∼6	10%
6∼8	30%
8∼10	9%
>10	1%

TABLE 3. Time spent on the proposed method and CAD modeling.

Images	Proposed method(s)	CAD modeling(s)
Figure. 14 (a)	1.317	300 ∼ 600
Figure. 14 (b)	1.631	
Figure. 14 (c)	1.550	
Figure. 14 (d)	1.633	
Figure. 14 (e)	2.482	
Figure. 14 (f)	5.347	

experiments, a sedan group, a SUV group and a sketch group. Fig. 13 shows some key points extraction results including sedan, SUV, and sketches.

B. COARSE MODEL GENERATION

Considering the main features of the side car body, the coarse extraction is only carried out on the outer contour of the car body, the characteristic line of the window and door, and the waistline. The simplified model is composed of 47 cubic Bézier curves, as seen from the coarse extraction results shown in Fig. 14.

C. FINE MODEL OPTIMIZATION

Combining the coarse extracted model in Figure.14 and the Bézier curve auto-fitting algorithm in Algorithm 2, a more accurate model will be reconstructed fitted to the edge information as shown in Fig. 15; the red line is the coarse extracted model, and the green line is the optimized accurate model.

In order to verify that the automatic model extraction method proposed in this paper is more efficient than the CAD modeling method, we invited 5 experienced engineers to model manually starting from the images in Fig. 13. The corresponding time spent is shown in Table 3, which demonstrates that the automatic car body modeling method proposed in this paper is efficient and fast, and can save a lot of time.

Further comparison results with the method in [27] illustrates that this method is more quickly and easily, the time consuming is reduced largely in this method in the order of 5 minutes less and the constraints before for the fine extraction is ignored here. The experimental results of the two methods are compared as shown in Fig. 16.

VII. CONCLUSION

This paper proposes a fast and efficient modeling method of 2D parametric car model. This method follows the idea of 'from coarse to fine' and 'knowledge reuse'. Firstly, few specific key points of the side view are automatically ''coarsely to finely'' extracted by a deep neural network, and then the improved OMP algorithm is used to extract a coarse model with the specific key feature points as input, combining the model database. Further, Bézier curve auto-fitting algorithm is proposed with the minimization idea that the bias length between the line points and the image edge points, then the optimal position of the control points of the cubic Bézier curve is found by iterative method. Finally, the objective feature curves are fitted to the real edge information combining the coarse model.

The experimental results show that the model extraction based on knowledge reuse can greatly shorten the design cycle. The running time of the Bézier curve auto-fitting algorithm is milliseconds, which can significantly improve the optimization efficiency.

This article takes sedan and SUV as examples to demonstrate the efficiency of car body automation modeling. For other car types, such as trucks, the method in this article can also be used, but it is necessary to establish a key point and model database of the truck in advance.

The model extraction accuracy is affected by the edge information, and a more ''cleaner'' edge can lead to a more accurate model. More training images can get better key point extraction results. Therefore, future works will be divided into two aspects: 1) more effective edge detection algorithms to reduce the extracted error; 2) improved training set with more car images to get better performance of the deep neural network for the key points extraction.

REFERENCES

[1] C. E. Catalano, F. Giannini, M. Monti, and G. Ucelli, ''A framework for the automatic annotation of car aesthetics,'' *Artif. Intell. Eng. Des., Anal. Manuf.*, vol. 21, no. 1, pp. 73–90, Jan. 2007, doi: 10.1017/S0890060407070151.

[2] B. Wang, ''An effective image reuse approach for product styling,'' *J. Inf. Comput. Sci.*, vol. 12, no. 1, pp. 235–246, Jan. 2015.

[3] D. L. Donoho, ''Compressed sensing,'' *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Sep. 2006.

[4] T. Blumensath and M. E. Davies, ''Gradient pursuits,'' *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2370–2382, Jun. 2008.

[5] S. G. Mallat and Z. Zhang, ''Matching pursuits with time-frequency dictionaries,'' *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[6] J. A. Tropp and A. C. Gilbert, ''Signal recovery from random measurements via orthogonal matching pursuit,'' *IEEE Trans. Inf. Theory.*, vol. 53, no. 12, pp. 4655–4666, Jan. 2008.

[7] D. L. Donoho, Y. Tsaig, I. Drori, and J. Starck, ''Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit,'' *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.

[8] D. Needell and R. Vershynin, ''Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit,'' *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 310–316, Apr. 2010.

[9] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, ''Active shape models-their training and application,'' *Comput. Vis. Image Understand.*, vol. 61, no. 1, pp. 38–59, 1995.

[10] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou, ''Look at boundary: A boundary-aware face alignment algorithm,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2129–2138.

[11] X. Guo, S. Li, J. Yu, J. Zhang, J. Ma, L. Ma, W. Liu, and H. Ling, ''PFLD: A practical facial landmark detector,'' 2019, *arXiv:1902.10859*.

[12] D. Qi, Z. Tian, Y. Zhang, and J. Zheng, ''The method of numeric Polish in curve fitting,'' *Acta Math. Sinica*, vol. 18, no. 3, pp. 173–184, 1975.

[13] H. Lin and Z. Zhang, "An extended iterative format for the progressive-iteration approximation," *Comput. Graph.*, vol. 35, no. 5, pp. 967–975, Oct. 2011.
[14] C. Deng and H. Lin, "Progressive and iterative approximation for least squares B-spline curve and surface fitting," *Comput.-Aided Des.*, vol. 47, pp. 32–44, Feb. 2014.
[15] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
[16] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid, "Groups of adjacent contour segments for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 1, pp. 36–51, Jan. 2008.
[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
[18] J. Kittler, "On the accuracy of the Sobel edge detector," *Image Vis. Comput.*, vol. 1, no. 1, pp. 37–42, 1983.
[19] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[20] V. Torre and T. A. Poggio, "On edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMT-8, no. 2, pp. 147–163, Mar. 1986.
[21] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 5, pp. 530–549, May 2004.
[22] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1558–1570, Aug. 2015.
[23] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3000–3009.
[24] R. Smith, R. Pawlicki, I. Kókai, J. Finger, and T. Vetter, "Navigating in a shape space of registered models," *IEEE Trans. Visualizat. Comput. Graph.*, vol. 13, no. 6, pp. 1552–1559, Nov./Dec. 2007.
[25] B. Li, Q. Li, C. Liu, M. Liu, P. Hu, and X. Liu, "Recovery of hierarchical styling features from images," *J. Inf. Comput. Sci.*, vol. 10, no. 7, pp. 1907–1916, 2013, doi: 10.12733/JICS2010165.
[26] J.-B. Bluntzer, E. Ostrosi, and J.-C. Sagot, "Car styling: A CAD approach to identify, extract and interpret characteristic lines," *Proc. CIRP*, vol. 21, pp. 258–263, Jan. 2014.
[27] B. Wang, F. Giannini, M. Monti, B. Li, P. Hu, and J. Liang, "Car model reconstruction from images through character line recognition," *Eng. Comput.*, vol. 35, no. 5, pp. 1873–1906, Sep. 2018, doi: 10.1108/EC-11-2016-0376.
[28] B. Wang, B.-J. Li, F. Giannini, M. Monti, P. Hu, and J.-C. Liang, "Curve-based image editing for product styling," *Comput.-Aided Des. Appl.*, vol. 15, no. 3, pp. 367–377, May 2018, doi: 10.1080/16864360.2017.1397887.

BO WANG received the Ph.D. degree from the Dalian University of Technology, Dalian, Liaoning, in 2017. She is currently a Lecturer with the Wuhan University of Science and Technology, Wuhan, China. She is mainly engaged in vehicle engineering teaching and scientific research. She presided over a National Natural Science Youth Foundation Project. Her main research interest includes car body CAD/CAS.

ZHIYING YANG received the bachelor's degree in vehicle engineering from the Wuhan University of Science and Technology, Wuhan, China, in 2019, where he is currently pursuing the master's degree in vehicle engineering. His current research interest includes car body CAD/CAS.

FRANCA GIANNINI received the degree from the Department of Mathematics, University of Genoa, in 1986. From November 1986 to February 1989, she was with the Research and Development Team, Italcad Tecnologie e Sistemi. Since 1989, she has been with IMATI-CNR, Genoa. She has been involved in some national and international projects on production automation, distributed product development, and the definition of user-friendly modeling tools. She is the coauthor of a patented system for automatic feature extraction (FEARE). Her research interests include product modeling, distributed design, and computer graphics.

MARINA MONTI received the degree in mathematics from the University of Genoa, in May 1984. Until 1997, she was with the companies Italcad and Computervision, in the Research and Development Department, working mostly in the fields of solid and free-form surfaces modeling and standards for the management of product data models. Since November 1998, she has been a part of IMATI-CNR, Genoa. She is involved in several international projects. Her research interests include computer-aided design, product modeling, and concurrent engineering.

BAOJUN LI (Member, IEEE) received the Ph.D. degree from the Dalian University of Technology, Dalian, Liaoning, in 2009. He is currently an Associate Professor with the Dalian University of Technology. He presided over and participated in more than 20 national, provincial, and ministerial research projects, including two 973 subprojects, two 863 subprojects, and seven National Natural Science Foundation Projects. He has published more than 70 academic papers in high-level academic journals and academic conferences, applied for and obtained more than ten national invention patents and appearance patents, and 12 software copyrights. His main research interests include big data-driven product intelligent design and integrated design, robotics and autonomous driving technology, and deep learning. He won the Second Prize in the Liaoning Province Natural Science Academic Achievement Award for two consecutive years, in 2011 and 2012.