Optimization and Engineering (2019) 20:683–723
https://doi.org/10.1007/s11081-018-9410-9

RESEARCH ARTICLE

A user-centered side silhouette generation system for sedan cars based on shape templates

Erkan Gunpinar[1] · Salih Ertug Ovur[2] · Serkan Gunpinar[3]

Received: 5 July 2018 / Revised: 11 November 2018 / Accepted: 11 November 2018 /
Published online: 22 November 2018

Abstract

A tool for the generation of side silhouettes for sedan cars is developed in this work; this enables ordinary people to produce personalized silhouettes that they prefer. Multiple silhouettes are collected from different car brands and analyzed in an off-line step. The silhouette is partitioned into nine characteristic lines, each of which is represented by either quadratic or cubic Bezier curves. Furthermore, geometric properties, such as geometric constraints and lower/upper bounds for the characteristic line dimensions, are determined. Next, a characteristic line is sampled via a spatial simulated annealing approach, while minimizing the Audze–Eglais potential energy between the sampled characteristic lines generated. These samples are called *line templates* or *shape templates*. A car side silhouette is encoded by a sequence of parameters, such as characteristic line lengths and heights and line template indices (i.e., 21 parameters in total). Once the values for these parameters are set, the silhouette can be generated. A user interface is provided for users to develop personalized car side silhouettes. According to the user study performed in this work, the developed tool enables users to obtain more desired/preferred silhouette(s) compared to those obtained using an interactive genetic algorithm.

Keywords Computer-aided design · Car side silhouette · User-centered design · Shape templates · Shape sampling

✉ Erkan Gunpinar
gunpinar@itu.edu.tr

[1] School of Mechanical Engineering, Istanbul Technical University, Inonu St. No: 65, Gumussuyu, 34437 Beyoglu, Istanbul, Turkey

[2] Istanbul Technical University, Istanbul, Turkey

[3] Dallas, TX, USA

1 Introduction

The importance of aesthetic contents in industrial design has been increasing, in addition to products price and quality. The current trend for customers is to choose products that not only fulfill functionalities, but also satisfy their inherently subjective tastes (Poirson et al. 2007). In the automotive industry, a car model's appearance is a decisive factor in attracting customers' attention. It has been demonstrated that the aesthetic aspects of a car amounts to 70% of customers' purchase intent (Cheutet 2007). Car design, development and production processes typically take more than 2 years. The first step begins with a simple, two dimensional car side silhouette sketch(es), which are drawn by a car stylist, and then these sketches are converted into three dimensional one(s). Computer renderings and clay models are then employed to visualize the concept as an physical object. Since the car stylist may not be aware of current market information, the sketch(es) obtained in the initial step might miss the target customers.

Collaborating with customers during the product design stage has a crucial role in market success. Existing works (Cluzel et al. 2012; Kelly et al. 2008; Tseng et al. 2012) in the literature mainly couple their tools with an interactive genetic algorithm (IGA) that can be used by customers for the generation of personalized car side silhouettes. IGAs allow people with little or no CAD experience to interact with software to create and modify designs and shapes (Tseng et al. 2012). However, one of the major limitations of IGAs as mentioned in Takagi (2001) is that as a human user cooperates with a indefatigable computer and evaluates individuals, thus IGA tends to produce fatigue after a few generations. The work in this paper introduces a different way for the car side silhouette generation to collaborate with customers. The tool developed is effective and easy-to-use with a higher design flexibility and enables customers to produce their preferred car side silhouettes. These silhouettes can be used as inputs in side silhouette generation by car stylists so that they can consider customer preferences while they draw the sketches. In addition, the tool provides opportunities for companies to effectively collaborate with their customers.

A side silhouette of a sedan car comprises of nine characteristic lines, as follows: the bumper line, grill line, hood line, windshield line, roof line, rear windshield line, two trunk lines and rear bumper line. Geometric constraints were identified by analyzing characteristic line shapes among the car side silhouettes of several car brands. In an off-line step, line templates for each characteristic lines were generated via a shape sampling approach while considering predetermined geometric constraints. The line templates were used in the car side silhouette generation step. Figure 1 shows a car side silhouette with the composing characteristic lines. The proposed system, which uses line templates, can be effectively and easily employed by any user after a short training period.

The remainder of this paper is organized as follows: Sect. 2 reviews relevant literature. Section 3 explains the proposed system for the generation of car side silhouettes. The results and discussion, and a user study are given in Sect. 4. Finally, concluding remarks and opportunities for future work are presented in Sect. 5.

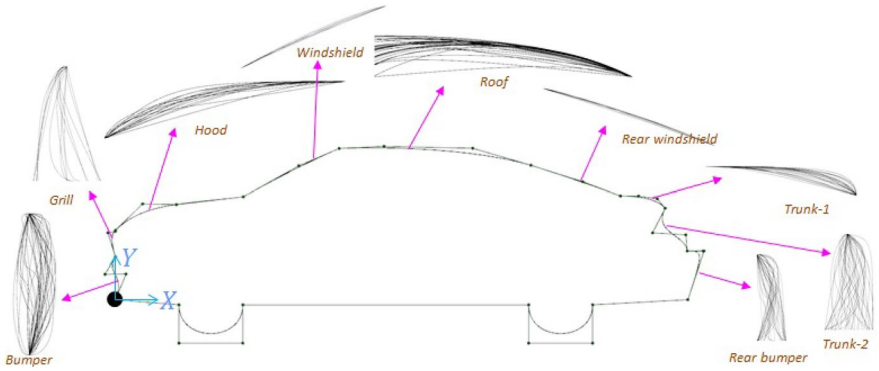


Fig. 1 A car side silhouette with nine characteristic lines having line templates shown next to them

2 Related works

The work reported here is related to the user-centered and car silhouette design concepts. Selected previous works are discussed in this section.

2.1 User-centered design

In user-centered design, the customer becomes an actor in the design process and his/her perceptions are considered with a view to improving the product. A considerable amount of work has been done on customer-centered design; only some of these studies are mentioned here. Poirson et al. (2007) implemented genetic algorithms for a user-centered design of products. Useful objective functions were deduced for the design of a new instrument, and the user-centered design problem was formulated as a multi-objective optimization problem. Moreover, Petiot and Yannou (2004) proposed a methodology for assessing product semantics; this is based on usability tests and involves several techniques, such as multidimensional scaling, the semantic differential method, and factor analysis. The tool they developed helps designers in understanding and specifying the semantic parts of the need. Furthermore, Giannini et al. (2006) introduced an approach in computer-aided industrial design to support users in attaining a model with a certain aesthetic and emotional character. Wu et al. (2009) developed an assistive device design process employing a user-centered design approach. In this work, an adjustable hair washer for physically disabled individuals was developed to wash their hair using normal postures. Hsiao and Huang (2002) proposed a neural network based approach for product design. Hsiao and Tsai (2005) proposed an automatic design system giving designers the ability to rapidly obtain a product form in a shorter time. Koc et al. (2008, 2009) introduced design principles demonstrating how different types of experience influenced people's knowledge of a product's use. A design tool was devised to implement these design principles. Hsiao et al. (2010) analyzed product for assessing product design feasibility styles by applying genetic algorithms and Kansei Engineering. There are recent works that can be employed for designing three dimensional car models.

Yumer and Kara (2014) proposed a deformation method that learns the implicit shape constraints in a product family from models such as cars. The handles allowed the user to free-form surface deformations. A hierarchical simplification of a product was produced, whereby customer response to geometric elements. Yumer et al. (2015) proposed a shape editing method to create geometric deformations using semantic attributes. This provided a design platform for quick design explorations. Khan and Gunpinar (2018) suggested CAD model sampling technique based on user preferences using an extended version of the teaching-learning-based optimization.

2.2 Car side silhouette design

The side silhouette of a car has a crucial role on the three-dimensional automobile (overall) form, and many researchers work on the silhouette during the conceptual design process. Here, we only elaborate the most relevant works. Tovey et al. (2003) outlined specific properties of automotive design sketches, such as form lines and crown lines. They confirmed the importance of the form lines in the design process, and thus the design of CAD systems to support concept development should consider these lines. Lai et al. (2005) suggested a design approach, which helps designers in enhancing the feeling quality of the products. A target feeling was specified and three shapes were redesigned based on the optimal parameters determined by the robust design. The approach was verified using a passenger car silhouette. Orsborn et al. (2006) introduced shape grammars to understand the differences between vehicle classes, and a class of vehicles was explored for unique designs. Mckay et al. (2006) incorporated evolutionary algorithms into a shape grammar-based design system to generate new product shapes. Furthermore, Aoyama et al. (2007) introduced four digital style design systems for cars to construct a shape from a rough sketch.

Cheutet (2007) proposed a semantic-based modeling environment for the conceptual design phase of a car. A car profile was represented using 10 lines, including the hood line, windshield line, roof line, wheelbase line, and wheel arch. A process grammar was integrated to describe and manipulate two dimensional (2D) sketches of the car. Orsborn et al. (2008) utilized principal component analysis to identify the fundamental characteristics within vehicle classes, which can be utilized to generate product specific forms. Orsborn et al. (2009) introduced a technique for quantifying a consumer's form preference in a utility function, which can be used to generate product forms based on the customer preferences. Kelly et al. (2008) studied the design of car silhouettes with interactive genetic algorithms, while Cluzel et al. (2012) proposed an interactive genetic algorithm to progressively sketch the desired car side silhouette, representing one of the most similar studies to the work in this paper. They adopted a Fourier decomposition of a 2D silhouette as the genotype and introduced a cross-over mechanism. Reid et al. (2010) proposed a method to assess subjective design attributes and developed a perceived environmental friendliness (PEF) model as a function of vehicle silhouette shape variables. Reid et al. (2012) further included empirically validated PEF silhouette attributes in a vehicle optimization model to maximize the fuel economy. Nam et al. (2013) proposed a spatial

modeling system for automotive design used in the conceptual modeling phase. The character lines were drawn with a curve-based sketch, and the car shape was modified using area-based modification. Hyun et al. (2015) synthesized car design alternatives using a methodology involving a mixture of Fourier decomposition, eye tracker and shape grammar while preserving brands design styles. Moreover, Hyun and Lee (2017) recently proposed a technique to synthesize car designs based on strategic styling positioning; their paper investigated ways of implementing design similarities for synthesizing styles. Shieh et al. (2017) proposed an integrated model based on multi-objective optimization and multi-criteria decision-making. A case study of car form design was conducted to demonstrate the proposed technique. Recently, Gunpinar and Gunpinar (2018) proposed a design sampling technique for CAD models to generate space-filling designs, in which a car side silhouette model was employed as a test case.

2.3 Contributions

The work in this paper focuses on the car side silhouette design from which car stylists begins with. The study involves an extensive shape manipulation algorithm, which samples shapes of characteristic lines such as bumper, hood, roof lines. The *design flexibility* of the proposed framework in terms of obtaining variety of characteristic line shapes is higher compared to those of the works (Cluzel et al. 2012; Yumer and Kara 2014; Yumer et al. 2015). For example, Cluzel et al. (2012) synthesized new silhouettes using a genetic algorithm, which begun with a set of silhouettes, and the new silhouettes were generated based on the user selections. Here, the initial population was given as input, which might not involve the silhouettes that the user like. Therefore, the algorithm might fail to generate the user preferred silhouettes. Additionally, Yumer and Kara (2014) and Yumer et al. (2015) utilized a given model as input, which was modified using handles. On the other hand, the proposed silhouette design framework can generate silhouettes with variety characteristic line shapes thanks to the sampling algorithm for the line templates.

The developed shape grammar by Orsborn et al. (2006). can be appropriate to be used by an automobile designer, however it looks that a novice user can not easily utilize the rules for the generation of the vehicle side view. Furthermore, the allowable ranges for the control points of the characteristic lines are rather narrow compared to those of our work. This may affect the generation of distinct characteristic lines. Smith et al. (2007) blended exemplars using wheelbase sliders to synthesize new silhouettes. The blending operation may restrict the design space exploration process as the synthesized silhouettes totally depends on the initial exemplars chosen. In our work, there is no need of exemplars. Distinct characteristic lines are sampled in advance, and new silhouettes can be generated via selecting line shapes and setting lengths/widths of the lines through horizontal sliders. Tseng et al. (2012) concurrently optimized stylistic form (such as sportiness, ruggedness, beautifulness and fuel efficiency) and functional requirements (such as aerodynamic performance). Rather than employing sliders to obtain desired silhouettes, an artificial neural network was facilitated to model consumer judgments of stylistic form quantitatively. Eight cubic Bezier curves and

two circles were utilized to represent the parametric car model, which was then simplified using 12 design parameters such as belt angle, nose angle, body height and hood length. Such simplified silhouette representation may not always describe a unique car side silhouette.

The main contributions of the proposed user-centered car side silhouette generation system are summarized in below:

1. A sampling algorithm for car silhouette characteristic lines in which Audze–Eglais potential energy between the generated lines are minimized. In this way, geometrically distinct characteristic lines can be generated. Such distinction can be favorable due to the following reasons:

   (a) With the distinct shapes for the characteristic lines, users can better scan the design space so that he/she can easily understand the possible shape options. Therefore, more preferred silhouettes can be generated.
   (b) In CAE tests such as car CFD simulations, designers tend to test distinct shapes to find out the optimum one based on the experiments conducted using distinct shapes.

   To the best of our knowledge, none of the previous works (such as Orsborn et al. 2006; Smith et al. 2007; Tseng et al. 2012) strove for the generation of characteristic lines that are distinct from each other as much as possible.
2. More desirable silhouette(s) can be generated using our silhouette generation system than those when using interactive genetic algorithms (i.e., Cluzel et al. 2012; Kelly et al. 2008) according to the user study conducted in this work. Second user study performed in this study reveals that the system is easy to use by ordinary people after spending a short warm-up time less than 5 min.

3 The proposed system

The proposed system includes three main steps, as shown in Fig. 2. In an off-line stage, shapes of characteristic lines are analyzed, and plausible shapes are determined. In this study, *plausible shapes* are the shapes satisfying the geometric constraints, which are learned in this shape analysis step. The upper and lower bounds of the characteristic line dimensions are also determined here. Using predetermined geometric constraints, characteristic lines are sampled, and each sample is called a *line template*. In this step, a fixed number of line templates are generated, which are as different from each other as possible in terms of appearance. In an online step, the car side silhouette is generated using line templates and upper/lower bounds for their dimensions.

3.1 Analyzing characteristic lines

In this step, we analyze characteristic lines for a set of sedan cars identified via company catalogs and the internet. Twenty-four car models were considered,

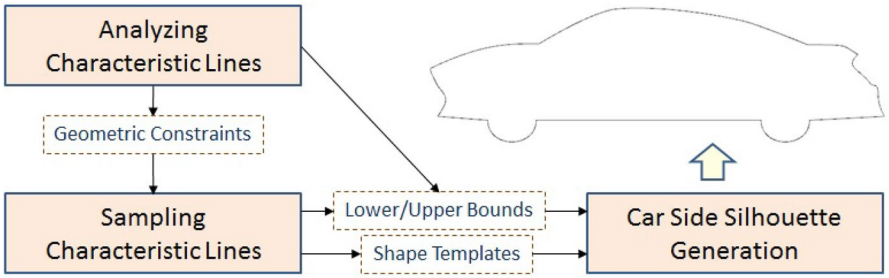


Fig. 2 System overview

representing some of the most popular cars on the market. Three models are from the Audi brand, two from BMW, two from FIAT, one from Ford, one from Honda, one from Hyundai, one from Mazda, one from Mercedes-Benz, one from Peugeot, three from Renault, two from Toyota, three from Volkswagen, and two from Volvo. Characteristic lines are represented using quadratic or cubic Bezier curves, as these are simpler and more efficient to use than higher degree curves. Quadratic Bezier curves are utilized for the front and rear windshield lines, and cubic Bezier curves are used for the others. Figure 3 shows a sedan car side view and its silhouette with control points. The silhouettes of the models are drawn one by one using a CAD tool. To analyze their dimensions, the overall sketch is scaled based on the real lengths of the models. The sketches for 24 Sedan car side silhouettes are shown in Fig. 4.

After obtaining Bezier curves representing characteristic lines, their control polygons are extracted. Sampling regions for the characteristic lines are then found, which are the minimum bounding rectangles or triangles of the control polygons oriented along the $X - Y$ axis. The sampling regions for the hood line, windshield and rear windshield lines are triangular, whereas those of other characteristic lines are rectangular. Orsborn et al. (2006) also employed sampling region (i.e., allowable range) for the control points to stay. However, the shapes of the sampling regions in our and their methods are quite different. They are either rectangular or triangular in our work, whereas the regions have polygonal shapes in Orsborn et al. (2006)'s work. The sampling regions we use are rather wider than those of their work, thus more distinct shapes of characteristic lines can be generated in our work. They employed separate sampling regions for the

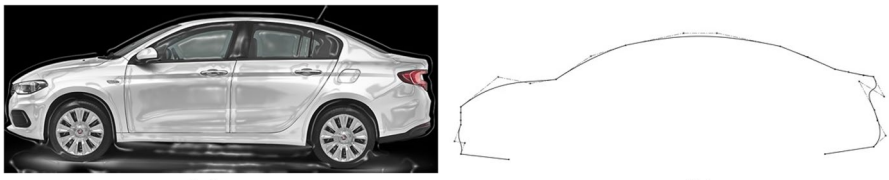


Fig. 3 Car side view (**a**) and its silhouette with control points (**b**)

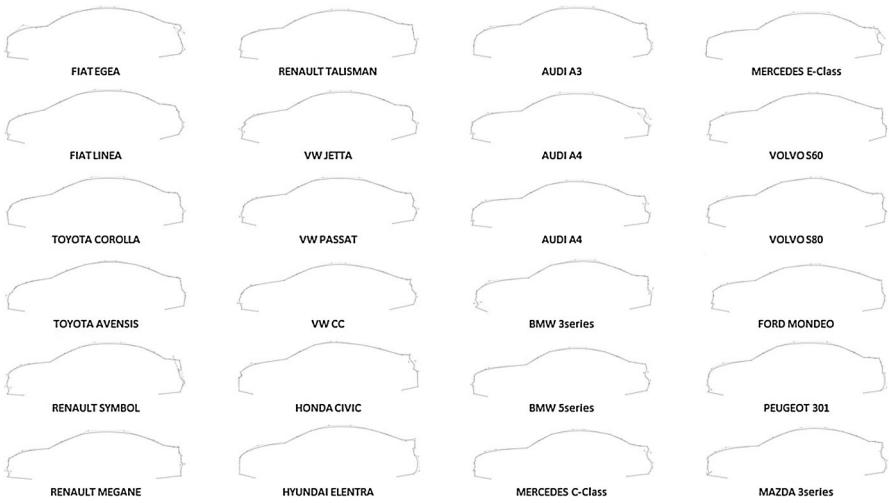


Fig. 4 Side silhouette sketches for twenty-four sedan car models

characteristic lines of coupes, pickups and SUVs. Note that the research objective of Orsborn et al. (2006)'s work is to develop shape grammars to quantify the differences between vehicle classes through the constraint of rule applications to within parametric ranges determined for each class. Therefore, they restrict the sampling region based on the vehicle classes. Such restriction is not preferable in our research as this will cause the generation of less distinct characteristic lines as our objective is to enable users to generate the silhouettes they like/prefer. Therefore, wider sampling regions are employed for the generation of characteristic lines. Pugliese et al. (2002) presented a shape grammar along with constraints for the Harley-Davidson brand. The allowable ranges for the design parameters are described as constraints (i.e., $62 < \iota < 70$ for the design parameter ι) rather than using sampling regions. Figure 5a illustrates a characteristic line (in blue) with its control polygon (in green). The sampling region for this line is shaded in gray. The length (l) and height (w) of the characteristic line are computed by subtracting the X and Y coordinate values of the end point from those of the start

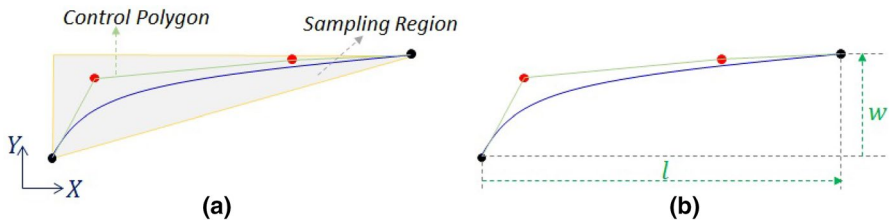


Fig. 5 **a** A characteristic line in blue with its control polygon in green is shown. Its sampling region is the triangle shaded in green. **b** The length and height of the characteristic line are the signed distance from its start point to the end point in the X and Y directions. (Color figure online)

point. Therefore, these parameters can obtain both positive and negative values. Figure 5b depicts the length and height of the characteristic line in blue.

Geometric properties of the characteristic lines are determined from the lines existing in the twenty-four sedan car models. First, characteristic line lengths and heights are measured one by one, and their lower/upper bounds are set. The maximum height and length of the sampling regions are also found using these car models. It should be noted that the sampling regions affect the formed shapes of the line templates in the sampling step. All these values are summarized in Table 1. Geometric constraints for the characteristic lines are then observed and identified. In fact, having too many geometric constraints is not desirable, as they will restrict the sampling process, causing less distinct characteristic lines to be generated. Therefore, only a few geometric constraints are determined and used in the characteristic line sampling process. Table 2 shows some shapes for the characteristic lines. According to our observations, shapes for the grill, hood, windshield, roof, rear windshield, and the first trunk lines do not have especially distinct shapes. However, other characteristic lines have diverse geometric shapes, and therefore, no geometric constraints are set for them. The following geometric constraints are mainly considered:

- The grill, hood, windshield, roof, rear windshield, and first trunk lines are represented using *slope-monotone descending curves* (Seong et al. 2002). In other words, these curves have continually decreasing slope; and
- The start and end angles for the roof line are less than 27° to avoid unusual characteristic line shapes.

3.2 Sampling characteristic lines

Let a characteristic line, C, be represented using a $2D$ cubic Bezier curve, and let (p_1, p_2, p_3, p_4) denote its control points. A sampling region, S, for C is the region encompassing all possible control polygons of C. Let p_1 and p_4 be the fixed control points, and let p_2 and p_3 be the mobile control points which can move within S. Figure 6a depicts fixed (in black) and mobile (in red) control points. While positioning mobile points to different locations in the sampling region, different line templates can be obtained, as seen in Fig. 6b. If a start or an end control point of the characteristic line is mobile, it moves up and down or left and right (see Fig. 6c). All inner control points for the characteristic lines are mobile. Furthermore, start control points for the grill and roof lines and end control points for the second trunk and rear bumper lines are given as mobile. Other control points are assigned as fixed points. Recall that characteristic curves are represented using either quadratic or cubic Bezier curves. Cubic curves are employed to explain the method; a similar technique is applied for the characteristic lines represented using quadratic curves.

A spatial simulated annealing (SSA) technique (Chen et al. 2013) is extended for sampling the line templates. A sampling number, Y, is given as input to the algorithm, which is the number of line templates generated by the SSA technique. The objective is to generate Y line templates, $(C_1, C_2, \ldots, C_Y)$, that are different from each other in terms of appearance and satisfy the predefined geometric constraints.

Table 1 Geometric properties of the characteristic lines

Characteristic line	Length (mm)		Height (mm)		Sampling region (mm)		
	Lower bound	Upper bound	Lower bound	Upper bound	Max. height	Max. length	Shape
Front bumper	–	–	220	450	450	209	Rectangular
Grill	–	–	110	300	300	180	Rectangular
Hood	850	1430	190	340	340	1430	Triangular
Windshield	580	930	320	400	400	930	Triangular
Roof	1320	1690	–	–	155	989	Rectangular
Rear windshield	540	860	210	290	290	860	Triangular
Trunk-1	220	520	–	–	100	520	Rectangular
Trunk-2	–	–	290	440	680	337	Rectangular
Rear bumper	–	–	320	490	490	166	Rectangular

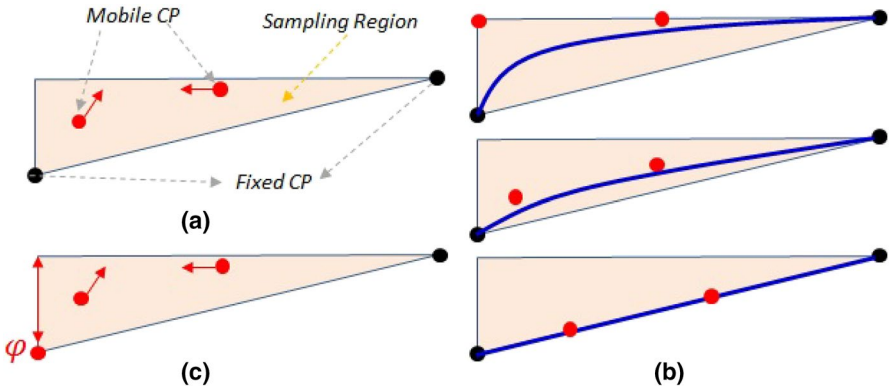


Fig. 6 **a** Fixed (in black) and mobile (in red) control points in the sampling region. **b** Line templates (in blue) generated using control points. **c** If the start control point φ of the characteristic line is mobile, it can move up and down (left and right in some cases). (Color figure online)

Table 2 Characteristic line shapes

Characteristic Line	Shapes
Front Bumper Line	
Grill Line	
Hood Line	
Front Windshield Line	
Roof Line	
Rear Windshield Line	
Trunk Line 1	
Trunk Line 2	
Rear Bumper Line	

The cost function F in Eq. 1 includes two terms and a coefficient, which must be minimized. α adjusts the weight between two terms.

$$F = E + \alpha * \overline{E}. \tag{1}$$

The term E favors the generation of line templates with different geometries. For this, we utilize a potential energy similar to those of Audze and Eglais (1977) and Fuerle and Sienz (2011), which is based on the analogy of minimizing forces

between charged particles. Particles—characteristic lines in this work— are in equilibrium in case of a minimum of the potential energy. E is defined as follows:

$$E = \sum_{p=1}^{Y} \sum_{q=p+1}^{Y} \frac{1}{D_{pq}^2} \tag{2}$$

D_{pq} denotes dissimilarity between the line templates C_p and C_q. To compute this, a *feature vector* is utilized, which is a set of features describing a line template and takes the form of a $3n \times 2$ matrix. Each feature (column) in this matrix represents a point or vector. First, n equally-spaced points on the line template are found. Moreover, $(v_1, v_2, \ldots, v_n)$ denote these n points. $(v'_1, v'_2, \ldots, v'_n)$ and $(v''_1, v''_2, \ldots, v''_n)$ are the first and second derivatives at the points, respectively. This data set forms a *feature space* of a characteristic line. The feature vector χ of the line template is written as

$$\chi = \begin{bmatrix} \chi_1 \\ \vdots \\ \chi_n \\ \chi_{n+1} \\ \vdots \\ \chi_{2n} \\ \chi_{2n+1} \\ \vdots \\ \chi_{3n} \end{bmatrix} = \begin{bmatrix} v_{1x} & v_{1y} \\ \vdots & \vdots \\ v_{nx} & v_{ny} \\ v'_{1x} & v'_{1y} \\ \vdots & \vdots \\ v'_{nx} & v'_{ny} \\ v''_{1x} & v''_{1y} \\ \vdots & \vdots \\ v''_{nx} & v''_{ny} \end{bmatrix} \tag{3}$$

Here, v_{ix} and v_{iy} are the x and y coordinates, respectively, of the point v_i, where $1 \leq i \leq n$. Similarly, v'_{ix}, v'_{iy}, v''_{ix} and v''_{iy} are the x and y coordinates of the vector denoting the first and second derivatives at v_i. Note that the computational cost of the proposed sampling technique is high when utilizing larger n values, as the feature vector contains more elements with $3n$ features. D_{pq} is computed based on the Euclidean distance between the feature vectors χ_p and χ_q of the line templates C_p and C_q, which is formulated as

$$D_{pq} = \sqrt{\sum_{j=1}^{3n} d^2(\chi_p^j, \chi_q^j)} \tag{4}$$

The function $d(\chi_p^j, \chi_q^j)$ is computed by subtracting jth feature of the feature vector χ_q from that of χ_p.

A static constraint handling mechanism (Homaifar et al. 1994) is utilized to penalize line templates that do not satisfy the predefined geometric constraints. The term $\overline{E}$ in the cost function F is used for this purpose. Suppose we have n_c

number of geometric constraints for a characteristic line and ϕ_t denotes a geometric constraint where $1 \leq t \leq n_c$. $\overline{E}$ is the total number of geometric constraint violations for each line template. Equation 5 is used to compute the value for the term $\overline{E}$. The function $f(\phi_t, C_p)$ has a value of zero 0 if the geometric constraint ϕ_t is satisfied by the line template C_p (see Eq. 6); otherwise, it is 1. Note that the parameter α in the cost function F is set to 10^{10} to give greater weight to the term $\overline{E}$, thereby favoring the generation of line templates satisfying the geometric constraints.

$$\overline{E} = \sum_{p=1}^{Y} \sum_{t=1}^{n_c} f(\phi_t, C_p) \tag{5}$$

$$f(\phi_t, C_p) = \begin{cases} 0 & \text{if } C_p \text{ satisfies } \phi_t \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

To sample characteristic line shapes, first, Y line templates are randomly produced. Line templates with implausible shapes are then determined and improved to make them plausible by randomly moving their mobile control points in the sampling region. Next, we apply an SSA approach for these feasible templates aiming to minimize the cost function in Eq. 1. In the SSA approach, the initial temperature T is set, and the algorithm stops when it becomes less than $\overline{T}$. S denotes the shortening rate for T. The Markov chain length L (i.e., iterative times at the same temperature level) is set to two times greater than the number Y of the line templates. A line template, C_w ($1 \leq w \leq Y$), is randomly selected, which is then moved in the sampling region by the distance H.

The mobile control point/points of C_w moves/move over a vector $\vec{h}$, with the direction chosen randomly. The length $\left|\vec{h}\right|$ of $\vec{h}$ takes a random value between 0 and H. H is shortened as the simulated annealing temperature decreases, and it is computed using the following formula: $H = H' \times S^c \times rand(0, 1)$, where S is the shortening rate and c is the number of drops in the SSA temperature. The function $rand(0, 1)$ produces a floating point number between 0 and 1. To compute H', a line is first created (see the dotted red line, Fig. 7a) starting from the control point, which is in the direction of the vector h. An intersection point (see the orange point) between this line and the sampling region boundary is found. H' is the distance between the intersection point and the control point. Figure 7b illustrates H and H'. Note that H is always smaller than or equal to H', as the shortening rate S is set to 0.9.

Let ζ be the plausible line template set (i.e., $C_1, C_2, \ldots, C_Y$). After the Markov chain (i.e., $2Y$ times line template generations), the line template set minimizing the cost function F, ζ_{best}, is selected. The temperature T is then decreased, and the new line template set is again searched. The pseudo-code in Algorithm 1 summarizes the SSA approach utilized in this work.

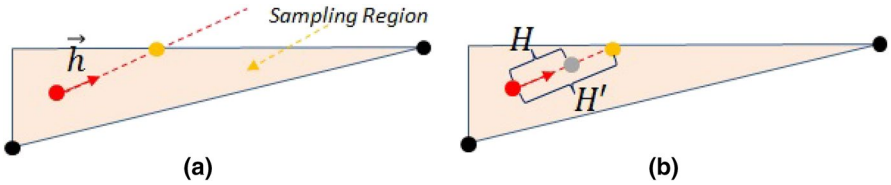


Fig. 7 **a** A control point (in red) of a line template can move in the direction of the vector $\vec{h}$. An intersection point (in orange) between the sampling region boundary and the line starting from the control point in the direction of $\vec{h}$ is computed. **b** The distance H, used to move the control point in the SSA approach, is based on H', which is the distance between the control point and the intersection point. (Color figure online)

Algorithm 1 The Spatial Simulated Annealing approach.

```
for T = 80; T ≥ 0.00008; T = T × S do
    Compute the cost F for ζ
    for i = 1 to 2Y do
        Randomly select a line template, C_w
        Move the mobile control point(s) of C_w within the sampling region
        Set the new line template set ζ′
        Compute the new cost F′ for ζ′
        if F′ ≤ F then
            Set ζ to ζ′ and F to F′
            Set ζ_best to ζ′ and F_best to F′
        else if F′ ≥ F and e^((ζ−ζ′)/T) ≤ rand(0, 1) then
            Set ζ to ζ′ and F to F′
        end if
    end for
    if F ≥ F_best then
        Set ζ to ζ_best and F to F_best
    end if
end for
```

3.3 Car side silhouette generation

A car side silhouette is represented using a chromosome, ψ, with 21 variables, as follows:

$$\psi = \left[t_1, w_1, t_2, w_2, t_3, l_3, w_3, \ldots, t_9, w_9\right]. \tag{7}$$

Here, t_i, l_i, and w_i represent line template, length, and height for the characteristic line, where $1 \leq i \leq 9$. Note that no length or height parameter is assigned to some of the characteristic lines, as they have negligibly small (i.e., visually hardly recognizable) lengths or heights. Therefore, the bumper, grill, trunk-2, rear bumper lines have negligible lengths, while the roof and trunk-1 lines have negligible heights. Table 3 shows the variables corresponding to the characteristic lines. The line templates are generated using the technique described in Sect. 3.2. They are then sorted based on the similarities (i.e., Euclidean distances in Eq. 4). To accomplish this, the

Table 3 Variables in the chromosome representing the car side silhouette

Characteristic line	Line template	Length	Height
Bumper line	t_1	–	w_1
Grill line	t_2	–	w_2
Hood line	t_3	l_3	w_3
Windshield line	t_4	l_4	w_4
Roof line	t_5	l_5	–
Rear windshield line	t_6	l_6	w_6
Trunk-1 line	t_7	l_7	–
Trunk-2 line	t_8	–	w_8
Rear bumper line	t_9	–	w_9

two most dissimilar line templates are first found, and the other templates are positioned between them according to distance proximity. As a result, the values for the line template variables change between integer values, from 1 to number of sampled line templates (i.e., indices in the sorted order). Length and height variables can be adjusted between lower and upper bounds for the lengths and heights of the characteristic lines, which are determined in Sect. 3.1.

3.3.1 Scaling line templates

Once the chromosome variable values of a car side silhouette are known, it is possible to generate the silhouette. It should be noted that the line templates are formed in the sampling region with a fixed length and height, therefore they need to be scaled based on the length and height values. The scaled characteristic line should be as visually similar to the chosen line template (i.e., the original model) as possible. Let a characteristic line, C, be represented by the control points (p_1, p_2, p_3, p_4). The objective is to find positions of the new control points $(\underline{p}_1, \underline{p}_2, \underline{p}_3, \underline{p}_4)$ for the characteristic line $\underline{C}$ so that the original (C) and scaled ($\underline{C}$) characteristic lines are similar. To compute similarity between two characteristic lines, feature vectors are first formed by neglecting the n top rows in Eq. 3 and using the rows between $n+1$ and $3n$. Equation 4 is then used to calculate the Euclidean distance between the two feature vectors, where j changes from $n+1$ to $3n$. We neglect the first n rows, as after scaling, it becomes less meaningful to compare the Euclidean distance between sampled points on the curve.

It is possible to search for a characteristic line, that is similar to the original characteristic line in the scaled sampling region using the SSA approach. However, it is impractical to use this technique in an interactive silhouette design stage as the search process takes time. Therefore, a more practical approach is proposed for scaling a given characteristic line, whose performance is validated in Sect. 4. Let l and w be the length and height, respectively, of C, where

$l = x_4 - x_1$ and $w = y_4 - y_1$. Moreover, $\underline{l}$ and $\underline{w}$ are the length and height of $\underline{C}$. In addition, x_i and y_i denote the X and Y coordinates of p_i, where $1 \leq i \leq 4$. Similarly, $\underline{x}_i$ and $\underline{y}_i$ are the X and Y coordinates of $\underline{p}_i$.

A similar approach to that of Khan et al. (2016) is utilized to update the control point positions, which is done to set the new length and height of the characteristic line. The new sampling region, $\underline{R}$, is first modified using the current length $\underline{l}$ and height $\underline{w}$. The first control point, $\underline{p}_1$, remains in its position, whereas the position of the last control point, $\underline{p}_4$), is modified using Eq. 8 (see Fig. 8b):

$$\underline{x}_4 = x_1 + \underline{l}, \quad \underline{y}_4 = y_1 + \underline{w} \tag{8}$$

To update the positions of the inner control points p_2 and p_3, we keep the tangent vectors same (see red and pink vectors in Fig. 8a) at the start and end points of the curve C. In other words, the inner control points ($\underline{p}_2$, $\underline{p}_3$) can be positioned along these tangent vectors. Let d_2 and d_3 be the start and end tangent vectors, respectively, for the characteristic line C. A *border point* is a point that the inner control point cannot exceed (see blue dots in Fig. 8c). To compute border points, two lines are formed; the line κ_2 passing through $\underline{p}_1$ in the direction of d_2, and the line κ_3 passing through $\underline{p}_4$ in the direction of d_3. Then, $\underline{p}_2^t$ denotes the border point for $\underline{p}_2$, which is the intersection point between κ_2 and the sampling region $\underline{R}$. Similarly, $\underline{p}_3^t$ is the border point for $\underline{p}_3$, which is computed by finding the intersection between κ_3 and the sampling region $\underline{R}$. The vectors $\underline{d}_2$ and $\underline{d}_3$ denote the vectors from $\underline{p}_1$ to $\underline{p}_2^t$ and $\underline{p}_4$ to $\underline{p}_3^t$, respectively. The candidate points $\underline{p}_2^c$ and $\underline{p}_3^c$ for $\underline{p}_2$ and $\underline{p}_3$ (see green and orange dots in Fig. 8c) are found, respectively, using Eqs. 9 and 10.

$$\underline{p}_2^c = \underline{p}_1 + \underline{d}_2 t_2 \qquad \text{where } 0 \leq t_2 \leq 1 \tag{9}$$

$$\underline{p}_3^c = \underline{p}_4 + \underline{d}_3 t_3 \qquad \text{where } 0 \leq t_3 \leq 1 \tag{10}$$

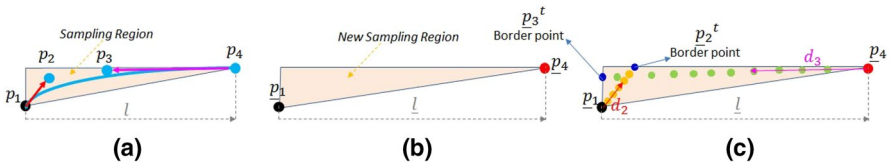


Fig. 8 Modification of lengths **a** and heights **b** of two cubic Bezier curves. Curves in blue are the original, and curves in red are the modified curves. (Color figure online)

Here, the parameters t_2 and t_3 vary between 0 and 1. The candidate point for $\underline{p}_2$ is $\underline{p}_1$ when t_2 is zero. It is the border point $\underline{p}_2^t$ for $t_2 = 1$. In addition, t_2 and t_3 are set to a set of values, $0, 1/\iota, 2/\iota, \ldots, 1$, where ι is an integer and is set to 10 in this study's experiments. By doing this, equally spaced candidate points can be generated between $\underline{p}_1$ and $\underline{p}_2^t$, and $\underline{p}_4$ and $\underline{p}_3^t$. After determining the candidate points, we form the curve for each inner control point pair (i.e., $\underline{p}_2^c$ and $\underline{p}_3^c$) with the start and end control points ($\underline{p}_1$, $\underline{p}_4$). The curve that exhibits the most similarity to the initial curve (C), is selected and regarded as the scaled curve $\underline{C}$. Note that there is only one inner control point for the windshield and rear windshield lines. In these cases, candidate points for the inner control point are given as the union of the points $\underline{p}_2^c$ and $\underline{p}_3^c$, computed using Eqs. 9 and 10.

3.3.2 Silhouette generation

Once the lengths and heights of the characteristic lines are set, they are joined to generate the car side silhouette. Figure 9 illustrates this process. Characteristic lines are shown in black, and their control points and control polygons are shown in dark green color. Let C^1 denote the characteristic curve for the bumper line with the control points $(p_1^1, p_2^1, p_3^1, p_4^1)$, after setting its length and height. Similarly, C^2–C^9 stand for the other eight characteristic curves. Table 4 summarizes these notations for each characteristic line. For the car side silhouette generation, we first insert the bumper line (C^1) and make its first control point coincident with the origin (see black point in Fig. 9), without applying any rotations. Therefore, a translation is applied to the

Table 4 Symbols for characteristic lines and their control points

Characteristic line	Symbol	Control points
Bumper line	C^1	$p_1^1, p_2^1, p_3^1, p_4^1$
Grill line	C^2	$p_1^2, p_2^2, p_3^2, p_4^2$
Hood line	C^3	$p_1^3, p_2^3, p_3^3, p_4^3$
Windshield line	C^4	p_1^4, p_2^4, p_3^4
Roof line	C^5	$p_1^5, p_2^5, p_3^5, p_4^5$
Rear windshield line	C^6	p_1^6, p_2^6, p_3^6
Trunk-1 line	C^7	$p_1^7, p_2^7, p_3^7, p_4^7$
Trunk-2 line	C^8	$p_1^8, p_2^8, p_3^8, p_4^8$
Rear bumper line	C^9	$p_1^9, p_2^9, p_3^9, p_4^9$

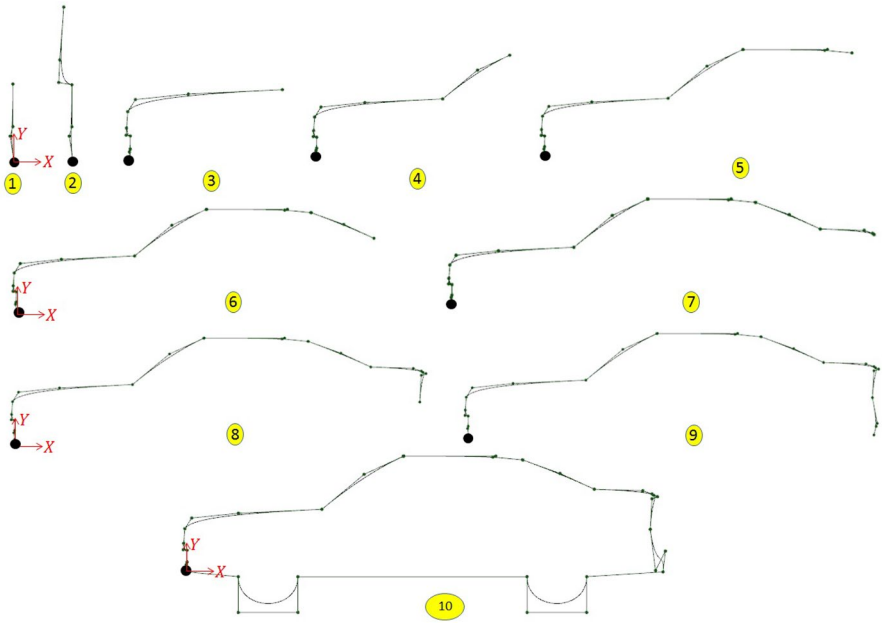


Fig. 9 Generation of a car side silhouette

control points of C^1, as follows: $\underline{C}^1 = C^1 * T(-p_1^1)$. $\underline{C}^i$ represents the modified ith characteristic line with its control points $(\underline{p}_1^i, \underline{p}_2^i, \ldots; (\underline{x}_1^i, \underline{y}_1^i), (\underline{x}_2^i, \underline{y}_2^i), \ldots))$. Here, $T(-p_1^1)$ is the translation applied for C^1, so that each control point is moved by $-x_1^1$ in the X direction and $-y_1^1$ in the Y direction, where x_1^1 and y_1^1 represent the X and Y coordinates of p_1^1. The grill line C^2 is next moved, and its first control point, p_1^2, is superposed on the last control point, $\underline{p}_4^1$, of $\underline{C}^1$. The translation $T(\underline{p}_4^1 - p_1^2)$ moves the control points of C^2 by $(x_4^1 - x_1^2)$ and $(y_4^1 - y_1^2)$ in the X and Y directions, respectively. Other characteristic lines (C^3–C^9) are moved similarly and their modified versions are denoted by $\underline{C}^3$–$\underline{C}^9$. The pseudo-code below (Algorithm 2) explains the translation strategy for the characteristic lines.

Algorithm 2 Transformation of the characteristic lines.

1: Translate C^1 to the origin $\Rightarrow \underline{C}^1 = C^1 * T(-p_1^1)$
2: Join C^2 with $C^1 \Rightarrow \underline{C}^2 = C^2 * T(\underline{p}_4^1 - p_1^2)$
3: Join C^3 with $C^2 \Rightarrow \underline{C}^3 = C^3 * T(\underline{p}_4^2 - p_1^3)$
4: Join C^4 with $C^3 \Rightarrow \underline{C}^4 = C^4 * T(\underline{p}_4^3 - p_1^4)$
5: Join C^5 with $C^4 \Rightarrow \underline{C}^5 = C^5 * T(\underline{p}_3^4 - p_1^5)$
6: Join C^6 with $C^5 \Rightarrow \underline{C}^6 = C^6 * T(\underline{p}_4^5 - p_1^6)$
7: Join C^7 with $C^6 \Rightarrow \underline{C}^7 = C^7 * T(\underline{p}_3^6 - p_1^7)$
8: Join C^8 with $C^7 \Rightarrow \underline{C}^8 = C^8 * T(\underline{p}_4^7 - p_1^8)$
9: Join C^9 with $C^8 \Rightarrow \underline{C}^9 = C^9 * T(\underline{p}_4^8 - p_1^9)$

The lower part of the car side silhouette is composed of three linear Bezier curves and two cubic Bezier curves representing the wheel portions. Its length is equal to the overall length $\underline{x}_4^9 - \underline{x}_1^1$ of the car side silhouette. The front linear curve starts from the origin (i.e., the start point $\underline{p}_1^1$ of the bumper line), and the back linear curve should end at the end point $\underline{p}_4^9$ of the rear bumper line. In this way, the characteristic lines can form a plausible car side silhouette (i.e., closed). The front and back linear curves are slightly inclined to have a realistic look, whereas the middle linear line is parallel to the X-axis. The lengths of the linear lines change proportionally to the overall length. In contrast, the dimensions of the wheel portions do not change.

3.3.3 Chromosome repairing

When the described car side silhouette generation technique is applied, the silhouettes obtained looks implausible in some cases. Therefore, chromosome repair or healing is essential. The first case is depicted in Fig. 10, where the rear bumper line is below the $Y = 0$ line after joining the characteristic lines at each other's end. The problem arises from the height variables $(w_1, w_2, w_3, w_4, w_6, w_8, w_9)$ in the chromosome. Let Δ^w denote the sum of the characteristic line heights, which is computed as follows: $\Delta^w = w_1 + w_2 + w_3 + w_4 + w_6 + w_8 + w_9$. Δ^w should be close to zero to have a car side silhouette with a natural look; therefore, Δ^w is considered to be zero.

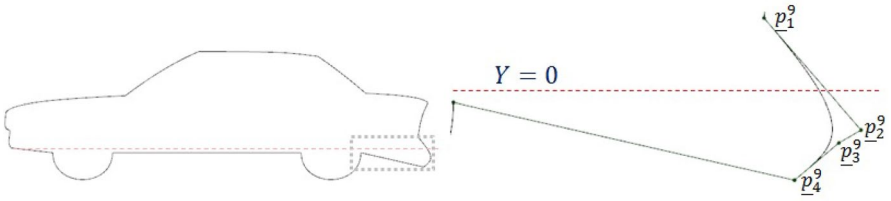


Fig. 10 Rear bumper line is below the $Y = 0$ line, which is unusual for a car side silhouette

$|\Delta^w|$ represents the sum of the absolute characteristic line heights, which is as follows: $|\Delta^w| = |w_1| + |w_2| + |w_3| + |w_4| + |w_6| + |w_8| + |w_9|$. The characteristic line heights are updated using Eq. 11. Here, $\underline{w_i}$ is the updated height of the ith characteristic line ($i = 1, 2, 3, 4, 6, 8, 9$). This formula proportionally adjusts the heights of characteristic lines. Once the updated heights are calculated, the characteristic lines are scaled using the strategy described in Sect. 3.3.

$$\underline{w_i} = w_i - \left(\frac{\Delta^w}{|\Delta^w|} * |w_i| \right) \tag{11}$$

Another chromosome healing can be done based on the user demand to increase the aerodynamic performance of the car side silhouette and to obtain a silhouette with a more plausible look. There is no smooth connection (i.e., no G^1 continuity) at the connection point between the windshield line and roof line, as seen in Fig. 11a, which increases the air drag aerodynamic drag on the silhouette. The same situation exists for the roof line and rear windshield lines for the same reason. Therefore, a simple

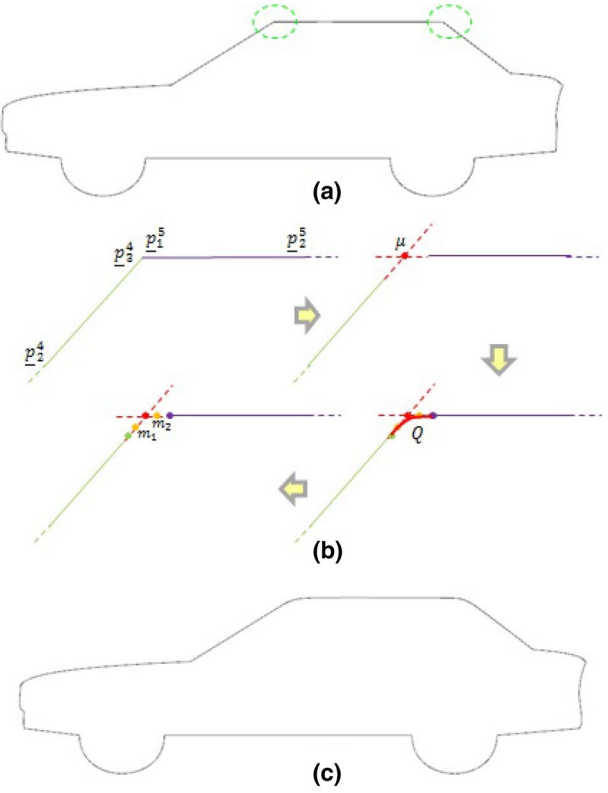


Fig. 11 **a** A car silhouette with no smooth connections at the start and end points of the roof line. **b** Chromosome repairing procedure. **c** Car side silhouette after repairing its chromosomes

heuristic is used consisting of four steps, which can be employed if desired. Algorithm 3 summarizes the technique and adds a smooth transition between the windshield and the roof lines. Let Ψ^4 and Ψ^5 be the curve length of the windshield line $\underline{C}^4$ and roof line $\underline{C}^5$, respectively. We first scale down the characteristic lines, and their curve lengths become $(\Psi^4 - \beta)$ and $(\Psi^5 - \beta)$, where β is set to 100, giving a more plausible look to the silhouette. An intersection point, μ, between the line connecting the control points $\underline{p}_2^4$ and $\underline{p}_3^4$ and the line connecting the control points $\underline{p}_1^5$ and $\underline{p}_2^5$ is found. Let m_1 and m_2 denote the mid-points between μ and $\underline{p}_3^4$, and μ and $\underline{p}_1^5$, respectively. A quartic curve, Q, is formed between the updated windshield and roof lines, where their control points are $\underline{p}_1^5$, m_1, μ, m_2 and $\underline{p}_3^4$. Due to the chromosome healing, the length parameters $(l_3, l_4, l_5, l_6, l_7)$ and height $(w_1, w_2, w_3, w_4, w_6, w_8, w_9)$ variables in the chromosome are repaired. A similar approach is used to smooth the transition between the roof and rear windshield lines. Figure 11b illustrates the process of adding a smooth transition between curves, and the produced car side silhouette is shown in Fig. 11c.

Algorithm 3 Adding a smooth transition between the windshield and roof lines.

1: Scale down $\underline{C}^4$ and $\underline{C}^5$
2: Compute the intersection point μ
3: Calculate the points m_1 and m_2
4: Form a quartic curve, Q, between the windshield and the roof lines, where their control points are $\underline{p}_1^5$, m_1, μ, m_2 and $\underline{p}_3^4$

3.3.4 Overall geometric constraints

The developed system warns the user if a geometric constraint is violated. Geometric constraints can differ for cars from different brands. In this work, only two constraints are utilized, as follows:

- *Overall length:* The overall length L (in mm) of the car side silhouette is restricted between lower and upper values, namely $4400 \leq L \leq 5500$. This limitation is determined based on the A, B, C, D types of sedan cars; and
- *Engine space:* There must be enough height (in mm) to mount the engine under the roof line. Therefore, the following limitation is considered: $(\underline{w_1} + \underline{w_2}) \geq 300$.

3.3.5 User interface

The car side silhouette generation system is easy to utilize; its user interface is depicted in Fig. 12. The system has been implemented as a plug-in to Openflipper (Mobius and Kobbelt 2012, a geometry processing and rendering framework) and

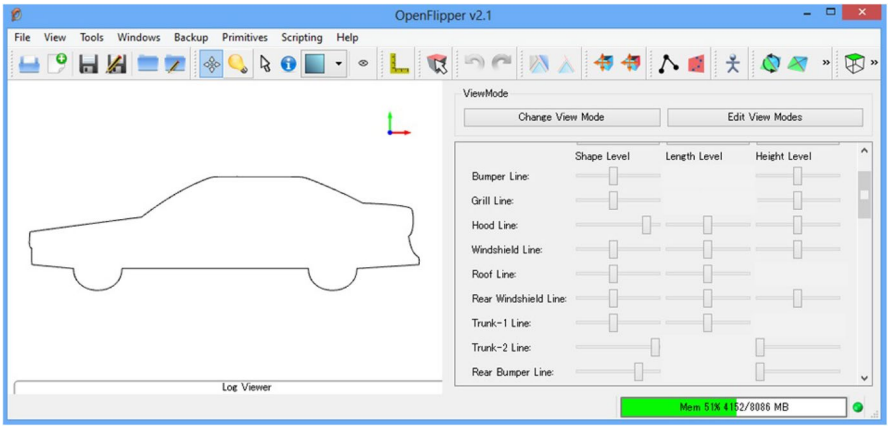


Fig. 12 User interface of the car side silhouette generation system

coded in Visual Studio 2008 using the Openflipper's API functions. Given a silhouette, the user only needs to adjust the sliders shown on the right, and the silhouette is immediately updated in the visualization frame on the left. There are three types of sliders, as follows: one for the line templates (in the sorted order based on the curve similarity explained in Sect. 3.3) of the characteristic lines, and the others for the lengths and heights of the characteristic lines. The lower and upper bounds of the characteristic line lengths and heights are obtained when the sliders are positioned at the left- and right-most positions, respectively.

4 Results and discussion

Line templates are sampled via the SSA technique, as shown in Fig. 1. Different shapes are generated in the sampling region of the characteristic line while satisfying the geometric constraints. In the SSA approach, the initial temperature T is set to 80.0, and the algorithm stops when it is less than $\overline{T}$, which is set to 10.0 for the experiments. The shortening rate S for T is set to 0.9. It is possible to assign greater values (but less than 1.0) for this parameter, which increases the computational time.

4.1 Computational time

A PC with an Intel Core *i*7 4820 3.7 GHz processor and 8 GB memory was used for the experiments in this study, and the implementation was single-threaded. Table 5 shows the computation time taken for sampling the line templates, which changed from 0.2 to 2696.0 s when the number of line templates Y to be sampled was set between 5 and 40. The time taken to sample 40 bumper line templates was 2696.0 s, which was more than two times longer than that (1102.5 s) needed to sample 30 bumper line templates. As a result, the processing time mainly depends on the parameter Y. The main reason for this is the Markov chain length L, which is set

Table 5 Computational time in seconds for sampling the line templates

Characteristic line	Computational time
Bumper ($Y = 30$)	1102.5
Bumper ($Y = 40$)	2696.0
Grill ($Y = 15$)	137.5
Hood ($Y = 15$)	139.4
Windshield ($Y = 5$)	2.0
Roof ($Y = 15$)	141.2
Roof ($Y = 30$)	1148.8
Rear Windshield ($Y = 5$)	0.2
Trunk-1 ($Y = 20$)	334.4
Trunk-1 ($Y = 30$)	1133.5
Trunk-2 ($Y = 40$)	2655.6
Rear bumper ($Y = 40$)	2609.0

to two times greater than Y. When the same number of different line templates were sampled, the processing times were similar. Sampling the bumper, second trunk and rear bumper lines was completed between 2600 and 2700 s. Finally, the time taken for the generation of car side silhouette was less than 1 s.

4.2 Parameter tuning

Several parameter-tuning experiments were carried out to obtain a better understanding of the parameters presented in this paper. The shortening rate S influences the decrease amount in T and the move amount H of the mobile control points. In other words, a higher value for S results in a lower decrease in T and higher moving distance H in each iteration. Table 6 shows the energy and computation time for the sampled grill line templates when Y is 15. Setting S to higher values produces samples with lower potential energies and increased in computational time.

The amount of decrease in T was also tested for the roof line, with $Y = 30$, and the rear bumper line, with $Y = 20$. The values of S were set to 0.9 and 0.9999 for the former and latter lines, respectively. Figure 13a shows a graph of the roof line, and convergence occurs after an approximately 25 times decrease in T. In case of the rear bumper line, the energy exhibits stable values after about 50 times decrease in T as seen in Fig. 13b.

Table 6 Shortening rate (S), energy $\overline{E}$ and computational time (t) in seconds for the sampled grill line templates ($Y = 15$)

S	0.9	0.99	0.999
$\overline{E}$	0.000294485	0.0000853456	0.0000191177
t	142.039	1442.1	13984.3

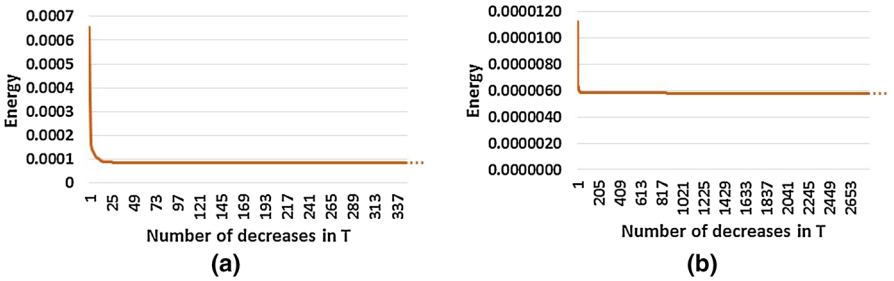


Fig. 13 Number of decreases in T versus energy ($\overline{E}$) for the roof line with $Y = 30$ (**a**) and the rear bumper line with $Y = 20$ (**b**)

4.3 Validation of the scaling algorithm

We validated the scaling algorithm described in Sect. 3.3 visually. Figure 14 shows a bumper line (a) and hood line (b), where the input characteristic lines are highlighted with yellow arrows. The bumper line is scaled down to half its size in length and height. Moreover, it is scaled separately in the X, and then Y, directions. Similarly, the hood line is scaled down to 80% of its size in length and height. It is then scaled separately in the X, and then Y, directions. Figure 14 depicts the resulting characteristic lines after scaling (see characteristic lines next to the input characteristic lines). According to our observations, the scaled curve resembles the original curve visually. The time taken to scale these curves is negligible; therefore the proposed scaling algorithm is practical for use in the interactive car side silhouette generation step.

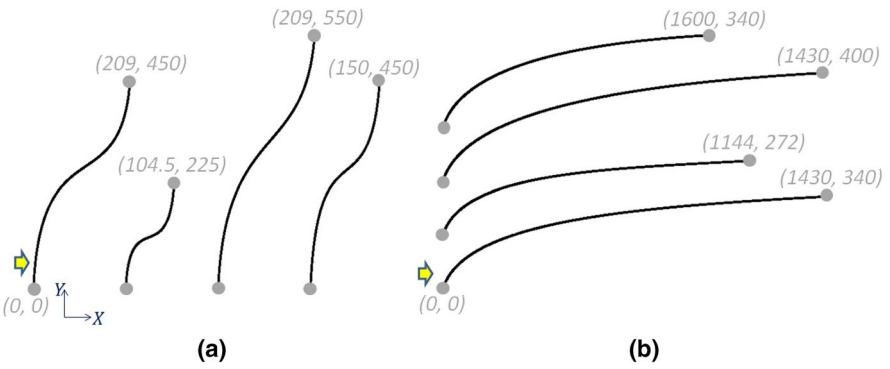


Fig. 14 Input characteristic lines (bumper line (**a**) and hood line (**b**)) indicated with yellow arrows are scaled by changing their lengths and heights. Scaled characteristic lines are illustrated next to the input characteristic lines. (Color figure online)

4.4 Producible models when using the line templates

Figure 15 shows the car side silhouettes that were generated using the user interface of the proposed work. Silhouettes with a variety of shapes can be generated by users when the line templates sampled in the initial step are utilized. Short or long, low or high silhouettes, compact silhouettes, and silhouettes resembling the silhouettes in Fig. 4 can be generated, as shown in Fig. 15. Some of these silhouettes have geometries far from or close to the streamlined geometry. It can be seen that the design space for the silhouettes in Fig. 4 is expanded as fewer geometric constraints are employed in the line templates' sampling step. If more geometric constraints are utilized, it may be hard to achieve such distinct shapes for the car side silhouettes. However, using the sampling technique with a less number of constraints can result in the generation of characteristic lines, some of which may be implausible for the

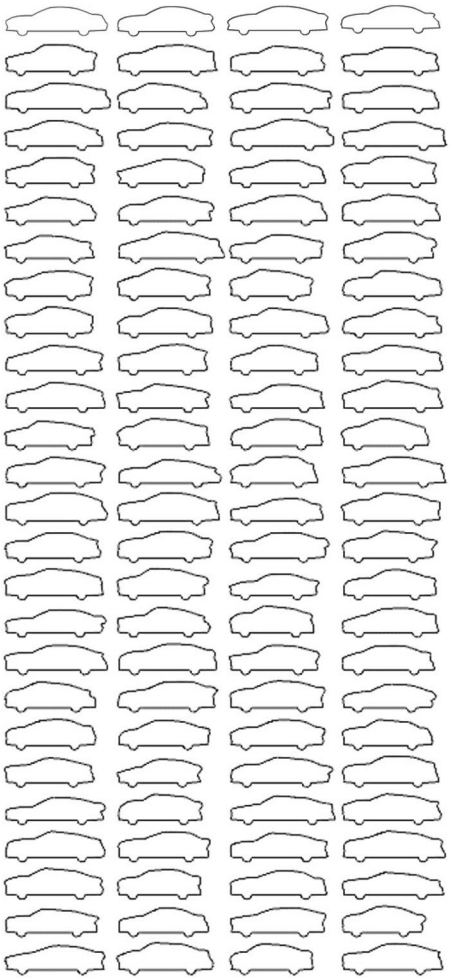


Fig. 15 Car side silhouettes generated using the user interface shown in Fig. 12

designers or users (see some of the trunk lines for the silhouettes in Fig. 15). In such cases, the elimination of undesirable characteristic lines may help to obtain more plausible silhouettes for the user. Therefore, one can only utilize some of the sampled characteristic lines in the car side silhouette generation system to obtain more plausible silhouettes.

The car side silhouettes generated using the proposed approach can be utilized by designers in a further design stage if desired. To do this, one approach involves obtaining a simplified CAD model for the silhouette, which is generated by sweeping the silhouette in the Z direction and then performing a cut sweep using the blue sketch on the $X-Y$ plane (see left image of Fig. 16). Figure 16 depicts three simplified CAD models (in brown) for the silhouettes shown at the top.

4.5 User study

A user study was conducted highlight the efficacy of the car side silhouette generation (CSSG) system. Two subject groups were involved the study. The first one (Group A) involves 14 users, who have professional design experience in industry and are capable of using SolidWorks or CATIA. The other group (Group B) have 36 users, who are ordinary people without any professional design experience. The subjects in the group A work for a company and enroll in a PhD program of Istanbul Technical University (ITU). The subjects in the group B are undergraduate students in Manufacturing Engineering Department of ITU. Small gifts were given to the subjects in exchange for their participation to the user study. The group A and B have, respectively, one and six female subjects, while others are male. Age average and standard deviation of the group A is 27.3 ± 1.5, whereas it is 23.8 ± 1.25 for the group B. Furthermore, years of professional design experience for the group A is 2.86 ± 1.3 (average $\pm$ standard deviation). Table 7 in Appendix shows the demographics of subjects in the groups A and B. Before using IGA-1, IGA-2 and the CSSG system, approximately 3–5 min warm-up time was given to the subjects in order to make them to get used to these tools. Explanations about the tasks and usage of the tools were less than 10 min. Additionally, possible silhouettes that can be generated using the CSSG system were shown to the subjects before implementing the user study.

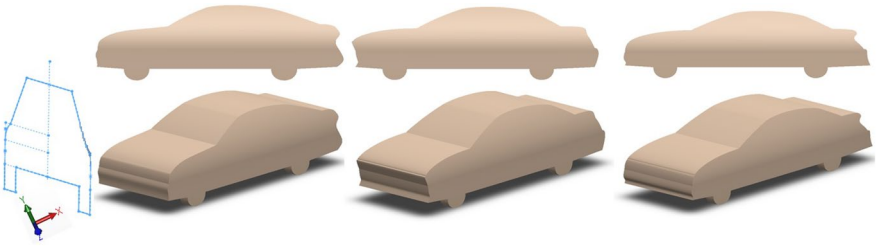


Fig. 16 Simplified CAD models (in brown) for the car side silhouettes are generated after sweeping the side silhouette in the Z direction and performing a cut sweep using the blue sketch on the $X-Y$ plane. (Color figure online)

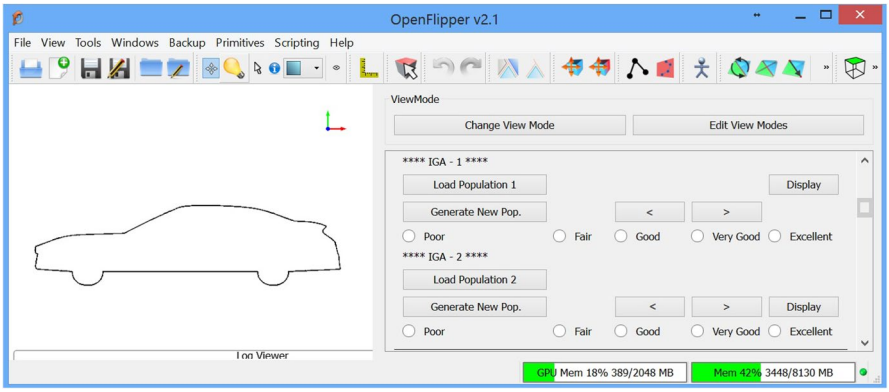


Fig. 17 User interface for the silhouette scoring system (IGA-1 and IGA-2)

Three tasks were given to the subjects. In the first and second tasks (IGA-1 and IGA-2), two different silhouette populations were shown to the subjects as depicted in Fig. 17. The subjects were then scored the silhouettes using a Likert scale (1: Poor, 2: Fair, 3: Good, 4: Very good, 5: Excellent) based on his/her preference. An interactive genetic algorithm (IGA) like that of Cluzel et al. (2012) was next utilized to obtain the new car side silhouettes (i.e., new population) based on the subject preferences. Subjects were again graded the new silhouettes separately using the Likert scale. The silhouettes with higher scorings were finally shown to the subjects, who chose three among them. In the third task, subjects were asked to generate a three car side silhouettes based on their preference using the CSSG system (see Fig. 12). Finally, the all nine models obtained from the three tasks were scored at the same time using the above-mentioned Likert scale.

4.5.1 IGA employed in the user study

IGA performance is directly related to the initial population used. Therefore, we tested its performance using two different silhouette populations, which are shown in Fig. 18a, b. In IGA-1, there were 27 silhouettes, two of which were duplicated for checking user reliabilities. However, there were 22 silhouettes in IGA-2 and similarly two of them were duplicated. These silhouettes were shown to the users for the evaluation. The silhouettes preferred were used to generate a new population in the evolution step, which were again shown to the users for the further selection. The population size was set to 40 here. 70% of the silhouettes were generated using a single-point crossover mechanism and 30% of them were obtained using a mutation operator.

To obtain a child silhouette using crossover, tournament selection was used to determine the parents. We randomly picked five silhouettes among the mating pool and the silhouette with the highest scoring became a parent. The tournament repeated for every parent required. If there were two parents having same scores, one of them was randomly chosen as a parent. Let S_1 and S_2 be the two parents that were chosen in the tournament selection. A crossover point were then randomized

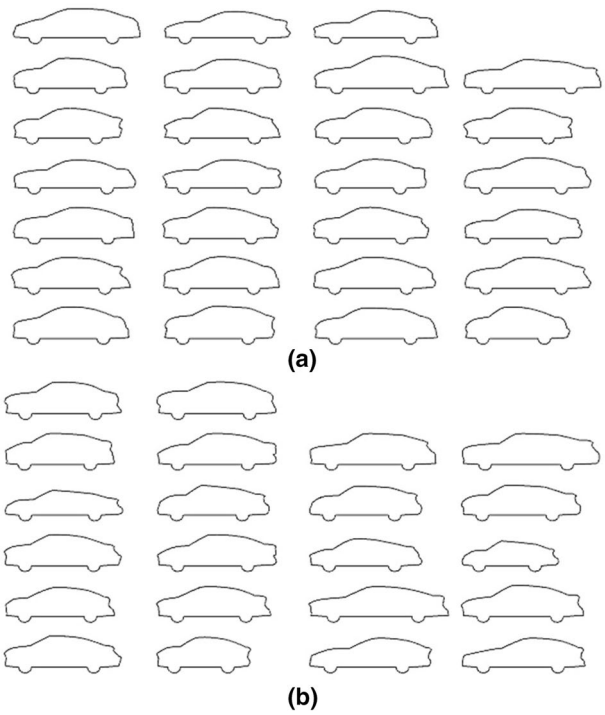


Fig. 18 Initial silhouette populations used in IGA-1 (**a**) and IGA-2 (**b**)

between 1 and 21 in each crossover operation as a silhouette consists of 21 chromosomes. Let x_R be the crossover point and an integer. A child was obtained by combining the first x_R chromosomes of S_1 and the other $21 - x_R$ chromosomes of S_2. To mutate the silhouettes, we randomly chose chromosomes from the silhouettes and combined them to obtain a new silhouette.

4.5.2 User reliability

When subject ratings are subjective or qualitative, special care is required for crowdsourcing user studies (Kittur et al. 2008). In this work, unreliable responses were filtered out using the following two techniques as done in Yumer et al. (2015):

- **Consistency** For the first and second tasks, an initial population and a new population generated by IGA consist of silhouettes. To check subject's consistency, we duplicated the silhouettes in every ten silhouettes. As a result, there were two same silhouettes in the initial silhouette population. For the new population generated using the above IGA technique, 40 new silhouettes were obtained. The population size was increased by adding four more silhouettes in every ten silhouettes, which were same as the ones in the population. Let Ω^1 and Ω^2 denote the consistency score of a subject for his selections in IGA-1 and IGA-2, respec-

tively. And Ω_s^1 and Ω_s^2 are the differences in the scores over 5 (i.e., 5−point Likert scale) given for the duplicated silhouette s, respectively, in IGA-1 and IGA-2. Note that there are six duplicated silhouettes. The consistency score Ω for a subject is computed as follows:

$$\Omega = \min(\Omega^1, \Omega^2) \tag{12}$$

$$\Omega^1 = 100 - \left(100 * \frac{\sum_{s=1}^{6} \Omega_s^1}{6}\right) \tag{13}$$

$$\Omega^2 = 100 - \left(100 * \frac{\sum_{s=1}^{6} \Omega_s^2}{6}\right) \tag{14}$$

Consistency score of a subject in IGA-1 or IGA-2 is computed by finding average of the differences in the scores given for the duplicated silhouettes. The consistency score is expressed in percentiles and calculated using Eqs. 13 and 14. The overall consistency score Ω of a subject is the minimum of Ω^1 and Ω^2 (see Eq. 12). Responses of subjects with consistency scores less than 80% were not used in the evaluation stage.

- **Diligence** Subjects complete the tasks in a short time were discarded. We discarded the participants who complete the task less than a threshold of seconds, which was computed by subtracting the standard deviation of the task completion time from the mean completion time. The threshold values were 102 and 52 s, respectively, for the tasks using IGA and CSSG.

4.5.3 Evaluation of the user study

As a result of the reliability test, the responses of the subjects, whose user numbers are 5, 11, 19, 22, 23, 24, 27, 29, 33, 34, 36, 37, 40, 44 and 48, were removed from the subject pool as seen from Table 8a in Appendix. We evaluated the results based on the time taken for the tasks and subject's scoring. Table 8b depicts the evaluation results for each participant. After excluding unreliable responses, average scores and their standard deviations were computed. The silhouettes generated using the CSSG system had a score of 3.69 ± 0.67 and 3.83 ± 0.57 (i.e., mean $\pm$ standard deviation) for the participants in the groups A and B, respectively. The participant scores were lower for the silhouettes generated using IGA-1 and IGA-2. Scores of IGA-1 were 3.28 ± 0.59 and 3.54 ± 0.72 for the subjects in the groups A and B, whereas they were 3.28 ± 0.81 and 3.19 ± 0.75 for IGA-2. Survey results are also analyzed using Box and Whisker plots (see Fig. 21a), which visually represents spread of data distribution.

We have also analyzed the statistical performance for the (whole) user scores (of the groups A and B) of the CSSG system over those of IGA-1 and IGA-2. The Shapiro-Wilk and Kolmogorov-Smirnov normality tests were performed to check

the distribution of the data in Table 8b. The significance values for these tests were smaller than 0.05, which imply that the data have a non-normal distribution. The histogram plot for the results in Table 8b is shown in Fig. 19a also confirmed this. Therefore, the data have a non-normal distribution and the Friedman test (Sheldon et al. 1996), a nonparametric statistical test, was used.

In the Friedman test, the null, *H0*, and alternative, *H1*, hypothesis are defined. The null hypothesis states that there is no significant difference between the methods, whereas the alternative hypothesis anticipates the presence of a significant difference between the methods (Derrac et al. 2011; Khan and Gunpinar 2018). χ^2 and p values provide information about whether a statistical hypothesis test is true or not. If the p value is less than 0.05 and the value of χ^2 is greater than the critical value of χ^2, then there is strong evidence against H0.

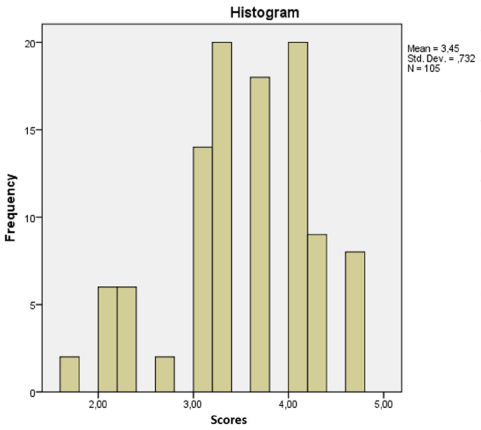
We performed two types of Friedman test using the datasets: multiple and pairwise comparison. The multiple comparison examined the multiple methods at once. On the other hand, individual comparisons between two different methods were performed in the pairwise comparison in which each method was compared to all other methods. The p and χ^2-values were 15.167 and 0.0005089, respectively, for the multiple comparison. And the critical χ^2 was 5.99 for the data set. As the p values were less than 0.05, and the χ^2 values were greater than the critical χ^2, it can be concluded that there is a strong evidence against H0. Figure 19b shows the p and χ^2 values for the pairwise comparison. When these values for CSSG and IGA-1/IGA-2 are compared, p values were less than less than 0.05, and the χ^2 values were greater than the critical χ^2 (3.84). As a result, the null hypothesis *H0* is rejected for this analysis as well. These findings reveal that the results shown in Table 8 (b) obtained using the CSSG system are significantly different from those obtained using IGA-1 and IGA-2 at alpha levels of 0.05 and less than 0.001, resp. Furthermore, medians for the user scores are 3.3, 3.3 and 4.0, respectively, for IGA-1, IGA-2 and the CSSG system. Consequently, it can be concluded that the CSSG system had statistically better performance (i.e., their models were scored better) than IGA-1 and IGA-2.

Additionally, Table 8a shows the interaction times with the IGA-1, IGA-2 and the CSSG system to generate each three silhouettes. To generate a single silhouette, the mean/standard deviation/median of the interaction times with IGA-1, IGA-2 and the CSSG systems are 239.8/84.5/224, 236.3/97.6/210 and 123.8/71.4/95, resp. Figure 20 shows the silhouettes obtained by the 3rd, 17th and 26th subjects using IGA-1, IGA-2 and the CSSG system.

4.5.4 User study II and its evaluation

After participants' short interaction with the CSSG system, we asked them for further evaluations about the following three questions:

- Q1. The CSSG system is easy to use for the generation of car side silhouettes.
- Q2. The CSSG system helps me to obtain car silhouettes that I imagine/like.
- Q3. Which system do you prefer for the generation of silhouettes?



(a)

p-value			
	IGA-1	IGA-2	CSSG
IGA-1	NaN	0.1444	0.04986
IGA-2	0.1444	NaN	0.00016
CSSG	0.04986	0.00016	NaN

$Critical\ \chi^2 = 3.84$

χ^2			
	IGA-1	IGA-2	CSSG
IGA-1	NaN	2.1304	3.8462
IGA-2	2.1304	NaN	14.286
CSSG	3.8462	14.286	NaN

(b)

Fig. 19 **a** Histogram plot for the data in Table 8b. **b** p and χ^2 values for the data in Table 8b obtained from the Friedman test in pairwise comparison

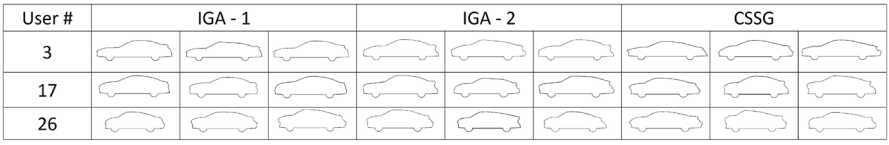
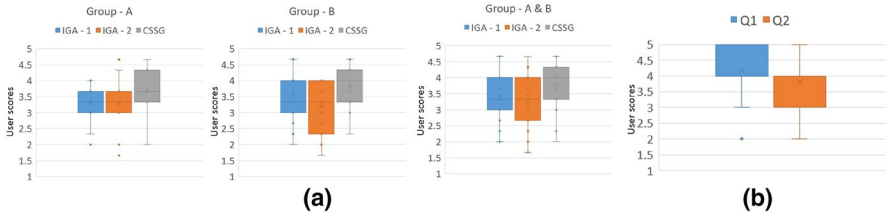


Fig. 20 The silhouettes obtained by the 3rd, 17th and 26th participants using IGA-1, IGA-2 and the CSSG system

The evaluations were performed based on a Likert scale (1: Strongly Disagree, 2: Disagree, 3: Neutral, 4: Agree, 5: Strongly Agree) and can be seen in Table 8c. The average scores and their standard deviations for the first and second questions are, respectively, as follows: 4.05 ± 0.81 and 3.88 ± 0.74. Survey results are also analyzed using Box and Whisker plots (see Fig. 21b). Furthermore, 92% of the subjects preferred to use the CSSG system rather than IGA. The reason to prefer the CSSG system was expressed by the participants that it had higher design flexibility so that more desirable design variations of silhouettes can be



(a) (b)

Fig. 21 **a** Box and Whisker plots for the user study evaluation in Table 8b. **b** Box and Whisker plots for user study II in Table 8c

obtained. Finally, some of the participants suggested to use the CSSG system with IGA. After finding a roughly preferred silhouette using IGA, the silhouette can be further improved via the CSSG system.

5 Conclusions and future works

This paper proposed a novel car side silhouette generation framework that can be used by ordinary people to develop personalized silhouettes. The car side silhouette comprised nine characteristic lines, and their geometric properties were learned by analyzing car silhouettes. The characteristic lines were then sampled via an SSA approach; the samples were called *line templates*. A car side silhouette was represented by a chromosome with 21 variables, such as characteristic line lengths, heights, and line template indices. A user interface allowed users to easily generate car side silhouettes.

In future work, we would like to extend our framework to 3*D* car models. Ways of using B-spline surfaces for the generation of new child models will be investigated. Furthermore, an adjective-based smart tool like in Dogan and Gunpinar (2017) will be developed to obtain attribute-based silhouettes such as big, small, fat, smart, high, long and short silhouettes. Additionally, a computational fluid dynamics-based silhouette optimization will be performed to obtain silhouettes while simultaneously optimizing their stylistic forms and aerodynamic abilities. Finally, the yacht hull design framework proposed by Khan et al. (2016) will be enhanced by means of 3*D* shape templates to involve ordinary people (i.e., customers) in the hull design process.

Acknowledgements The authors would like to thank Professor Ralph Martin for his helpful remarks, the Scientific and Technological Research Council of Turkey for supporting this research (Project Number: 315M077) and the RWTH Computer Graphics Group for making the OpenFlipper framework available.

Funding This study was funded by The Scientific and Technological Research Council of Turkey (Project Number: 315M077).

Compliance with ethical standards

Conflict of Interest We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome. We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us. We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property. We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author.

Appendix

See Tables 7 and 8.

Table 7 Demographics of the subjects

Groups	User #	Age	Gender	Years of pro. E.
A	1	27	M	3
	2	27	M	4
	3	28	M	4
	4	23	M	1
	5	28	M	4
	6	28	M	2
	7	27	M	4
	8	26	F	1
	9	28	M	2
	10	30	M	3
	11	28	M	4
	12	26	M	1
	13	28	M	5
	14	28	M	2
	Ave.	27.3	–	2.86
	SD	1.5	–	1.3
B	15	23	M	–
	16	24	M	–
	17	23	M	–
	18	25	F	–
	19	24	M	–
	20	23	M	–
	21	24	F	–
	22	23	M	–
	23	23	M	–
	24	22	M	–
	25	23	M	–
	26	24	M	–
	27	22	M	–
	28	24	F	–
	29	24	M	–
	30	22	M	–
	31	22	M	–
	32	23	M	–
	33	23	F	–
	34	23	M	–
	35	23	M	–
	36	23	M	–

Table 7 (continued)

Groups	User #	Age	Gender	Years of pro. E.
	37	24	M	–
	38	26	M	–
	39	25	M	–
	40	24	M	–
	41	24	M	–
	42	24	M	–
	43	27	F	–
	44	24	M	–
	45	23	M	–
	46	23	M	–
	47	23	M	–
	48	24	F	–
	49	28	M	–
	50	25	M	–
	Ave.	23.8	–	–
	SD	1.25	–	–

Table 8 (a) Reliability results. Ω^1, Ω^2: subject consistency scores, respectively, for IGA-1 and IGA-2 scorings, t_{iga1}, t_{iga2}: time in seconds taken, respectively, for IGA-1 and IGA-2 scorings, t_{cssg1}, t_{cssg2}, t_{cssg3}: time in seconds taken to generate each three silhouettes using the CSSG system, t_{cssg}: total time in seconds to generate three silhouettes via the CSSG system. (b) Subjects' scoring based on a Likert scale (1: Poor,..., 5: Excellent). η_1, η_2, η_3: score for the silhouettes, η_{ave}: average score for the silhouettes. (c) Results of user study II for three questions (Q1, Q2, Q3), (1: Strongly Disagree, 2: Disagree, 3: Neutral, 4: Agree, 5: Strongly Agree)

Group	User#	Ω^1	Ω^2	t_{iga1}	t_{iga2}	t_{cssg1}	t_{cssg2}	t_{cssg3}	t_{cssg}
(a)									
A	1	92	86.6	224	231	291	203	173	667
	2	100	95	304	347	99	63	139	301
	3	86.6	90	177	186	137	158	91	386
	4	90	88.3	337	229	260	229	149	638
	5	85	80	**90**	**48**	270	145	68	483
	6	83.3	80	275	292	172	115	71	358
	7	90	85	320	409	106	95	121	322
	8	90	90	176	190	90	110	70	270
	9	100	100	336	395	192	216	255	663
	10	90	85	542	554	125	115	90	330
	11	80	83.3	156	222	**46**	57	70	173
	12	90	86.6	226	245	82	70	68	220
	13	80	83.3	169	186	128	73	77	278
	14	93.3	90	409	440	115	232	200	547
B	15	90	83.3	224	222	105	92	95	292
	16	80	96.6	263	152	193	70	63	326
	17	90	90	215	214	79	56	58	193
	18	80	90	171	184	300	289	333	922
	19	85	93.3	**92**	118	64	54	**44**	162
	20	86.6	86.6	174	184	122	110	152	384
	21	90	96.6	183	189	125	95	102	322
	22	90	80	**77**	**62**	**39**	73	70	182
	23	85	**60**	222	242	162	135	98	395
	24	85	80	**92**	221	200	122	93	415
	25	80	100	218	170	117	135	110	362
	26	75	85	350	392	55	110	92	257
	27	**73.3**	100	123	120	77	65	68	210
	28	90	80	194	210	228	140	122	490
	29	**70**	90	**85**	**80**	125	105	121	351
	30	76.6	83.3	138	138	89	77	55	221
	31	90	100	246	173	276	225	140	641
	32	80	86.6	236	168	98	94	90	282
	33	90	83.3	**92**	**87**	**46**	**30**	**26**	**102**

Table 8 (continued)

Group	User#	Ω^1	Ω^2	t_{iga1}	t_{iga2}	t_{cssg1}	t_{cssg2}	t_{cssg3}	t_{cssg}
	34	100	85	**77**	**67**	82	90	97	269
	35	90	93.3	241	150	87	92	73	252
	36	83.3	80	142	164	54	**45**	**37**	136
	37	83.3	86.6	**45**	**70**	100	56	84	240
	38	86.6	80	225	185	145	158	98	401
	39	90	90	122	252	202	165	105	472
	40	83.3	**76.6**	226	164	58	**38**	**42**	**138**
	41	86.6	83.3	163	214	90	110	60	260
	42	83.3	85	251	197	180	187	125	492
	43	90	83.3	215	160	185	57	60	302
	44	90	90	**85**	**74**	92	88	60	240
	45	90	90	214	215	79	56	55	190
	46	80	85	105	106	74	85	82	241
	47	80	90	312	150	236	166	135	537
	48	85	**75**	**75**	**69**	70	52	101	223
	49	90	93.3	245	221	284	175	157	616
	50	83.3	90	192	319	71	65	68	204

Group	User #	IGA - 1				IGA - 2				CSSG			
		η_1	η_2	η_3	η_{ave}	η_1	η_2	η_3	η_{ave}	η_1	η_2	η_3	η_{ave}
(b)													
A	1	3	2	4	3	4	4	3	3.6	4	3	4	3.6
	2	4	4	4	4	3	4	3	3.3	4	4	3	3.6
	3	4	3	4	3.6	4	2	3	3	4	5	4	4.3
	4	2	4	3	3	4	2	3	3	3	3	4	3.3
	5	4	3	3	3.3	2	2	3	2.3	4	1	3	2.6
	6	3	4	5	4	5	5	4	4.6	5	4	5	4.6
	7	3	5	3	3.6	4	4	3	3.6	4	3	3	3.3
	8	1	2	3	2	1	2	3	2	2	3	1	2
	9	4	3	3	3.3	3	3	4	3.3	3	3	4	3.3
	10	4	3	3	3.3	4	2	3	3	5	4	4	4.3
	11	5	3	2	3.3	3	3	3	3	5	4	4	4.3
	12	5	4	2	3.6	4	5	4	4.3	5	4	2	3.6
	12	3	2	2	2.3	2	2	1	1.6	3	4	4	3.6
	14	3	3	4	3.3	4	3	4	3.6	4	4	5	4.3
	Ave.	–	–	–	3.28	–	–	–	3.28	–	–	–	3.69
	SD	–	–	–	0.59	–	–	–	0.81	–	–	–	0.67

Table 8 (continued)

Group	User #	IGA - 1				IGA - 2				CSSG			
		η_1	η_2	η_3	η_{ave}	η_1	η_2	η_3	η_{ave}	η_1	η_2	η_3	η_{ave}
B	15	4	3	5	4	3	5	4	4	3	4	5	4
	16	5	5	4	4.6	3	4	5	4	4	5	4	4.3
	17	4	3	3	3.3	3	5	4	4	4	4	5	4.3
	18	2	3	5	3.3	2	5	3	3.3	3	1	5	3
	19	3	4	4	3.6	4	3	2	3	2	4	5	3.6
	20	3	5	4	4	4	4	4	4	5	4	3	4
	21	4	3	4	3.6	2	3	3	2.6	3	4	4	3.6
	22	3	3	4	3.3	2	3	2	2.3	2	4	3	3
	23	4	3	4	3.6	4	4	4	4	4	4	5	4.3
	24	3	3	4	3.3	4	5	4	4.3	5	3	4	4
	25	3	4	3	3.3	3	2	1	2	3	5	2	3.3
	26	3	1	5	3	3	4	4	3.6	2	3	4	3
	27	2	3	1	2	4	3	2	3	4	3	2	3
	28	4	2	1	2.3	2	2	3	2.3	4	3	3	3.3
	29	5	4	3	4	2	3	2	2.3	4	4	3	3.6
	30	4	5	5	4.6	4	3	4	3.6	3	4	5	4
	31	3	2	4	3	2	2	1	1.6	5	4	4	4.3
	32	4	3	4	3.6	4	2	3	3	4	3	5	4
	33	4	4	3	3.6	5	3	2	3.3	4	2	2	2.6
	34	1	3	4	2.6	2	4	1	2.3	3	2	4	3
	35	2	4	3	3	3	3	3	3	3	4	4	3.6
	36	3	4	3	3.3	4	3	3	3.3	3	2	3	2.6
	37	4	4	3	3.6	4	5	5	4.6	4	5	5	4.6
	38	4	3	4	3.6	3	4	4	3.6	3	4	5	4
	39	2	2	2	2	1	2	3	2	4	4	2	3.3
	40	3	4	4	3.6	2	4	4	3.3	4	1	3	2.6
	41	4	3	3	3.3	3	3	4	3.3	4	5	5	4.6
	42	4	4	4	4	3	3	4	3.3	5	4	3	4
	43	3	3	2	2.6	3	2	2	2.3	3	3	4	3.3
	44	3	4	5	4	5	5	5	5	5	1	3	3
	45	4	3	3	3.3	3	5	4	4	4	4	5	4.3
	46	4	4	4	4	2	2	3	2.3	2	2	3	2.3
	47	5	5	4	4.6	4	4	4	4	5	4	4	4.3
	48	4	2	3	3	2	3	3	2.6	2	3	2	2.3
	49	5	5	4	4.6	3	2	4	3	4	4	4	4
	50	2	4	3	3	4	3	5	4	4	5	5	4.6
	Ave.	–	–	–	3.54	–	–	–	3.19	–	–	–	3.83
	SD	–	–	–	0.72	–	–	–	0.75	–	–	–	0.57

Table 8 (continued)

Group	User #	Q1	Q2	Q3
(c)				
A	1	4	4	CSSG
	2	4	4	CSSG
	3	5	5	CSSG
	4	4	5	CSSG
	5	5	4	CSSG
	6	5	4	CSSG
	7	5	3	CSSG
	8	4	2	CSSG
	9	5	3	CSSG
	10	3	4	CSSG
	11	5	4	CSSG
	12	4	3	IGA
	13	4	2	CSSG
	14	5	4	CSSG
	Ave.	4.43	3.64	–
	SD	0.62	0.89	–
B	15	5	5	CSSG
	16	4	4	CSSG
	17	3	4	CSSG
	18	5	5	CSSG
	19	2	3	CSSG
	20	4	3	CSSG
	21	4	4	CSSG
	22	3	3	CSSG
	23	5	5	CSSG
	24	5	4	CSSG
	25	3	3	CSSG
	26	4	5	IGA
	27	4	4	CSSG
	28	4	3	CSSG
	29	5	4	CSSG
	30	3	4	CSSG
	31	4	4	CSSG
	32	4	4	CSSG
	33	4	4	CSSG
	34	4	3	CSSG
	35	5	5	CSSG
	36	4	4	CSSG
	37	5	5	CSSG
	38	3	4	CSSG

Table 8 (continued)

Group	User #	Q1	Q2	Q3
	39	4	4	CSSG
	40	3	3	IGA
	41	4	4	CSSG
	42	5	4	CSSG
	43	4	4	CSSG
	44	5	5	IGA
	45	3	4	CSSG
	46	3	2	CSSG
	47	5	4	CSSG
	48	5	4	CSSG
	49	4	3	CSSG
	50	5	3	CSSG
	Ave.	4.05	3.88	–
	SD	0.81	0.74	–

References

Aoyama H, Nordgren A, Yamaguchi H, Komatsu Y, Ohno M (2007) Digital style design systems from concept to sophisticated shape. Int J Interact Des Manuf 1:55–65

Audze P, Eglais V (1977) New approach for planning out of experiments. Probl Dyn Strengths 35:104–107

Chen B, Pan Y, Wang J, Fu Z, Zeng Z, Zhou Y, Zhang Y (2013) Even sampling designs generation by efficient spatial simulated annealing. Math Comput Model 58(3–4):670–676

Cheutet F (2007) 2d semantic sketcher for a car aesthetic design. In: Proceedings of the CPI2007: Conception et Production Integrees

Cluzel F, Yannou B, Dihlmann M (2012) Using evolutionary design to interactively sketch car silhouettes and stimulate designers creativity. Eng Appl Artif Intell 25(7):1413–1424

Derrac J, Garcia S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evolut Comput 1:3–18

Dogan KM, Gunpinar E (2017) Learning yacht hull adjectives and their relationship with hull surface geometry using gmdh-type neural networks for human oriented smart design. Ocean Eng 145:215–229

Fuerle F, Sienz J (2011) Formulation of the audze-eglais uniform latin hypercube design of experiments for constrained design spaces. Adv Eng Softw 42(9):680–689

Giannini F, Monti M, Podelh G (2006) Aesthetic-driven tools for industrial design. J Eng Des 17(3):193–215

Gunpinar E, Gunpinar S (2018) A shape sampling technique via particle tracing for cad models. Graph Models 96:11–29

Homaifar A, Qi CX, Lai SH (1994) Constrained optimization via genetic algorithms. Simulation 62(4):242–253

Hsiao SW, Huang HC (2002) A neural network based approach for product form design. Des Stud 23:67–84

Hsiao SW, Tsai HC (2005) Applying a hybrid approach based on fuzzy neural network and genetic algorithm to product form design. Int J Ind Ergon 35:411–428

Hsiao SW, Chiu FY, Lu SH (2010) Product-form design model based on genetic algorithms. Int J Ind Ergon 40(3):237–246

Hyun KH, Lee JH (2017) Style synthesis based on strategic styling decision. In: Lee JH (eds) Morphological analysis of cultural DNA, KAIST Research Series. Springer, Singapore, pp 149–156. https://doi.org/10.1007/978-981-10-2329-3
Hyun KH, Lee J, Kim M, Cho S (2015) Style synthesis and analysis of car designs for style quantification based on product appearance similarities. Adv Eng Inf 29(3):483–494
Kelly J, Papalambros PY, Seifert CM (2008) Interactive genetic algorithms for use as creativity enhancement tools. In: AAAI spring symposium: creative intelligent systems, pp 34–39
Khan S, Gunpinar E (2018) Sampling cad models via an extended teaching-learning-based optimization technique. Comput Aided Des 100:52–67
Khan S, Gunpinar E, Dogan KM (2016) A novel design framework for generation and parametric modification of yacht hull surfaces. Ocean Eng 136:243–259
Kittur A, Chi EH, Suh B (2008) Crowdsourcing user studies with mechanical turk. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM, CHI '08, pp 453–456
Koc MC, Popovic V, Emmison M (2008) Using visual representation of concepts to explore users and designers' concepts of everyday products. Des Stud 29(2):142–159
Koc MC, Popovic V, Emmison M (2009) Human experience and product usability: principles to assist the design of user–product interactions. Appl Ergon 40(4):648–656
Lai HH, Chang YM, Chang HC (2005) A robust design approach for enhancing the feeling quality of a product: a car profile case study. Int J Ind Ergon 35(5):445–460
Mckay A, Chau MCAHH, Pennington AD (2006) Combining evolutionary algorithms and shape grammars to generate branded product design. In: Design computing and cognition '06. Springer, Dordrecht, pp 521–539
Mobius J, Kobbelt L (2012) Openflipper:an open source geometry processing and rendering framework. Springer, Berlin, Heidelberg, pp 488–500
Nam S, Patil AK, Patil S, Chintalapalli HR, Park K, Chai Y (2013) Hybrid interface of a two-dimensional cubic hermite curve oversketch and a three-dimensional spatial oversketch for the conceptual body design of a car. Proc Inst Mech Eng Part D J Automob Eng 227(12):1687–1697
Orsborn S, Cagan J, Pawlicki R, Smith RC (2006) Creating cross-over vehicles: defining and combining vehicle classes using shape grammars. Artif Intell Eng Des Anal Manuf 20(3):217–246
Orsborn S, Boatwright P, Cagan J (2008) Identifying product shape relationships using principal component analysis. Res Eng Des 18(4):163–180
Orsborn S, Cagan J, Boatwright P (2009) Quantifying aesthetic form preference in a utility function. ASME J Mech Des 131(6)::061001:1–061001:10
Petiot JF, Yannou B (2004) Measuring consumer perceptions for a better comprehension, specification and assessment of product semantics. Int J Ind Ergon 33(6):507–525
Poirson E, Depince P, Petiot JF (2007) User-centered design by genetic algorithms: application to brass musical instrument optimization. Eng Appl Artif Intell 20(4):511–518
Pugliese MJ, Michael J, Cagan J (2002) Capturing a rebel: modeling the harley-davidson brand through a motorcycle shape grammar. Res Eng Des 13(3):139–156
Reid TN, Papalambros PY, Gonzalez RD (2010) Quantification of perceived environmental friendliness for vehicle silhouette design. ASME J Mech Des 132(10):101010:1–101010:12
Reid TN, Frischknecht BD, Papalambros PY (2012) Perceptual attributes in product design: fuel economy and silhouette-based perceived environmental friendliness tradeoffs in automotive vehicle design. ASME J Mech Des 134(4):041006:1–041006:9
Seong JK, Kim MS, Sugihara K (2002) The minkowski sum of two simple surfaces generated by slope-monotone closed curves. In: GMP 02: proceedings of the geometric modeling and processing theory and applications (GMP02), Washington, DC, USA
Sheldon MR, Fillyaw MJ, Thompson WD (1996) The use and interpretation of the friedman test in the analysis of ordinal-scale data in repeated measures designs. Physiother Res Int 1:221–228
Shieh MD, Li Y, Yang CC (2017) Product form design model based on multiobjective optimization and multicriteria decision-making. Math Probl Eng. https://www.hindawi.com/journals/mpe/2017/5187521/
Smith RC, Pawlicki R, Kokai I, Finger J, Vetter T (2007) Navigating in a shape space of registered models. IEEE Trans Vis Comput Graph 13(6):1552–1559
Takagi H (2001) Interactive evolutionary computation: fusion of the capabilities of ec optimization and human evaluation. Proc IEEE 89(9):1275–1296
Tovey M, Porter S, Newman R (2003) Sketching, concept development and automotive design. Des Stud 24(2):135–153

Tseng I, Cagan J, Kotovsky K (2012) Concurrent optimization of computationally learned stylistic form and functional goals. ASME J Mech Des 134(11):1–11

Wu FG, Ma MY, Chang RH (2009) A new user-centered design approach: a hair washing assistive device design for users with shoulder mobility restriction. Int J Ind Ergon 40(5):878–886

Yumer ME, Kara LB (2014) Co-constrained handles for deformation in shape collections. ACM Trans Graph 33(6):187:1–187:11

Yumer ME, Chaudhuri S, Hodgins JK, Kara LB (2015) Semantic shape editing using deformation handles. ACM Trans Graph 34(4):86:1–86:12

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com