Data-Driven Synthesis of Cartoon Faces Using Different Styles

Yong Zhang, Weiming Dong, *Member, IEEE*, Chongyang Ma, Xing Mei, *Member, IEEE*, Ke Li, Feiyue Huang, Bao-Gang Hu, *Senior Member, IEEE*, and Oliver Deussen

***Abstract*— This paper presents a data-driven approach for automatically generating cartoon faces in different styles from a given portrait image. Our stylization pipeline consists of two steps: an offline *analysis* step to learn about how to select and compose facial components from the databases; a runtime *synthesis* step to generate the cartoon face by assembling parts from a database of stylized facial components. We propose an optimization framework that, for a given artistic style, simultaneously considers the desired image-cartoon relationships of the facial components and a proper adjustment of the image composition. We measure the similarity between facial components of the input image and our cartoon database via image feature matching, and introduce a probabilistic framework for modeling the relationships between cartoon facial components. We incorporate prior knowledge about image-cartoon relationships and the optimal composition of facial components extracted from a set of cartoon faces to maintain a natural, consistent, and attractive look of the results. We demonstrate generality and robustness of our approach by applying it to a variety of portrait images and compare our output with stylized results created by artists via a comprehensive user study.**

***Index Terms*— Cartoon face, face stylization, data-driven synthesis, component-based modeling.**

Manuscript received February 10, 2016; revised July 18, 2016 and October 14, 2016; accepted November 5, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61672520, Grant 61573348, Grant 61271430, and Grant 61372184, in part by the Beijing Natural Science Foundation under Grant 4162056, in part by the National Foreign Thousand Talents Plan under Grant WQ201344000169, in part by the Leading Talents of Guangdong Program under Grant 00201509, and in part by the CASIA Tencent Youtu Joint Research Project. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Catarina Brites. (Yong Zhang and Weiming Dong contributed equally to this work.) *(Corresponding author: Weiming Dong.)*

Y. Zhang is with the National Laboratory of Pattern Recognition and the Laboratory for Computer Science, Automation and Applied Mathematics, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yong.zhang@nlpr.ia.ac.cn).

W. Dong, X. Mei, and B.-G Hu are with the National Laboratory of Pattern Recognition and the Laboratory for Computer Science, Automation and Applied Mathematics, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: weiming.dong@ia.ac.cn; xing.mei@ia.ac.cn; baogang.hu@ia.ac.cn).

C. Ma is with the University of Southern California, Los Angeles, CA 90007 USA (e-mail: chongyang.ma@usc.edu).

K. Li and F. Huang are with the Youtu Laboratory, Tencent, Shanghai 200233, China (e-mail: keli@tencent.com; garyhuang@tencent.com).

O. Deussen is with the University of Konstanz, 78457 Konstanz, Germany, and also with the Shenzhen Institutes of Advanced Technology, Shenzhen 518172, China (e-mail: oliver.deussen@uni-konstanz.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2628581

I. Introduction

STYLIZED cartoon faces are widely used as virtual identities and personalized appearances in social media such as instant chat and online games. However, creating such images usually requires professional artistic skills and tedious manual work, which becomes impractical for many multimedia applications when there are millions or even billions of users. As a result, it is very useful to have automatic systems for cartoon stylization from portrait photos.

Traditional face stylization methods convert portraits into a specific style, such as line drawings [1], sketches [2], or exaggeration in terms of both shape and color [3]. However, it is not easy to generalize those frameworks to synthesize different stylization results of the same input. Furthermore, a typical face stylization method focuses on maintaining the similarity between the input portrait and the sketched or stylized output [4], but does not consider the proper arrangement of facial components as well as the interrelationships between them. This limitation usually leads to less attractive outputs, since human perception is especially sensitive to the appearance of faces [5]. On the other hand, previous approaches usually focus on stylizing faces in a specific style which often leads to the lack of generality to arbitrary style.

For a cartoon stylization system, the generated cartoon has to faithfully resemble the input portrait to keep the identity of the user. From an aesthetics point of view, the system needs also to provide visually pleasant and attractive abstractions of the input photo. Finally, an ideal approach should be able to efficiently generate stylized faces in arbitrary user-desired style with minimum modification to the system.

In this paper, we present a data-driven framework for automatic cartoon stylization of portrait photographs to address all the above challenges. As shown in Fig. 1, inspired by recent component-based shape modeling and synthesis techniques [6], [7], we first collect two databases of facial components (i.e., eyes, brows, nose, mouth, cheek and hair), see Section III-A. One of the databases contains realistic facial components collected from portrait photographs. The other database contains cartoon facial components of a certain style, which are drawn by professional artists to match the realistic components.

Next, we learn how to select cartoon components and assemble them together in an offline analysis stage. We exploit a Bayesian network to train a **selection** model for selecting cartoon components from multiple candidates to make an attractive face (Section III-B). We also adopt ε-SVR

Fig. 1. Given a portrait photo, we decompose the face into individual components and search for the corresponding cartoon candidates for each component in a database by feature matching. We then obtain the best combination of these candidates using a Bayesian network and compose the selected components together to synthesize a cartoon face. We can generate cartoon faces in different styles by our framework. The left is an example of *B&W* style and the right is an example of *FASHION* style.

(Support Vector Regression), cf. Chang and Lin [8], to train a **composition** model for optimally composing the selected cartoon facial components to the final cartoon (Section III-C).

In our synthesis stage (Section IV), we first segment the input portrait into facial components. For each component we compute specific features and then find the five most similar realistic components in the dataset by feature matching. We then generate several combinations of their corresponding stylized cartoon components and apply the trained Bayesian network to evaluate their attractiveness. The combination with the highest value is considered as the best output. Finally, we adjust the composition of the facial components to make the stylization results more natural and harmonic. We demonstrate the robustness, effectiveness and generality of our method through a variety of input portraits (Section V-A) and evaluate our method in a comprehensive user study (Section V-B).

In summary, our contributions are:

- a data-driven framework for component-based synthesis of stylized cartoon faces from input portrait photos;
- a learning algorithm to extract the information about how to select and compose components from a database of stylized facial components;
- an optimization-based method to generate a cartoon face by jointly optimizing photo-cartoon correspondence and the composition of different facial components.

Thus, compared with previous works which usually focus on maintaining the similarity between real and stylized faces in a specific style, our approach can balance the similarity to the input face and the visual attractiveness of stylized face in arbitrary user-desired style.

A preliminary extended abstract of this work appeared at Siggraph Asia 2014 [9].

II. Related Work

Creating artistic visual representations of human faces gained growing attention in computer graphics and computer vision. Related topics include face illustration [10], portrait painting [11], cartoonization [1], [12], [13], caricaturization [3], watercolorization [14], and replacement [15]. Since there are many related works, we focus on research in synthesizing personal appearances from photographs with an emphasis on data-driven approaches.

Synthesizing Facial Sketches: Chen et al. [1] present an example-based approach to generate cartoons from photographs by a non-parametric sampling scheme that captures the relationship between training images and corresponding sketches drawn by the same artist. This idea is further improved by performing the matching by a partial least-square technique using photo/caricature pairs [16], hierarchical-compositional models [17], feature-level nearest neighbor approaches [18], and semi-coupled dictionary learning from photo/sketch pairs [19]. Gooch et al. [10] create black-and-white facial illustrations from photographs and then deform these facial illustrations to create caricatures which highlight and exaggerate representative facial features. Min et al. [20] propose an automatic portrait system by leveraging the And-Or graph based on existing sketch templates. Wei et al. [21] use a statistical model to represent and to generate face wrinkles. Chen et al. [22] create face sketches by separating a face into individual components and recomposing them after transformation. Liu et al. [4] simulate artist's creativity on facial features and adopt a learning approach for mapping those for automatically creating caricatures. Tseng et al. [3] comprise a statistics-based exaggeration module and a non-photorealistic rendering module to create colored facial caricatures with exaggerated facial features. Tresset and Leymarie [23] present a robotic installation that produces sketches of people, but it is a "naive drawer" not capable of learning different styles. Berger et al. [24] gather and analyze data from a number of artists as they sketch a human faces from a reference photograph, and then use the trained models to synthesize portrait sketches. PortraitSketch system [25] assists users to draw sketches by adjusting their strokes in real time to improve their aesthetic quality. Patch-based methods are widely applied to synthesis of facial sketches due to their ability to represent local facial features [26]. Liu et al. [27] propose a face sketch generation method by employing the idea of locally linear embedding. Li et al. [12] generate cartoons by incorporating the content of guidance images taken from a specific training set. Gao et al. [28] improve the perceptual quality by compensating the high-frequency information and relaxing the number of the candidate image patches via sparse coding. Methods based on Markov random fields are proposed to create sketches from photos by selecting most appropriate neighbor patches to hallucinate a target patch [2], [29], [30]. Zhang et al. [31]

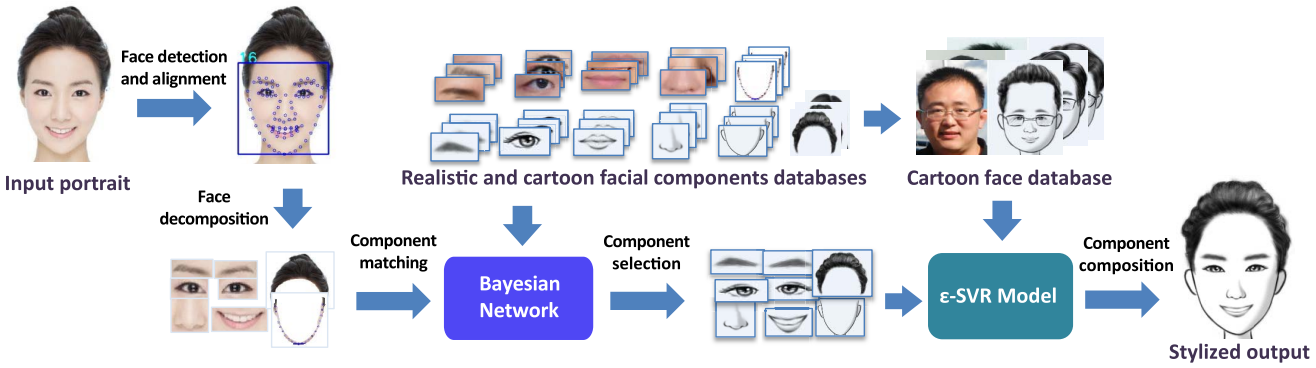


Fig. 2. The overall pipeline of our framework.

learn a feature dictionary from photo patches and replace these patches with a sparse parametric representation during the searching process. Zhang et al. [32] develop a framework to synthesize face sketches trained on only one template sketch use a multi-feature-based optimization model to select candidate image patches.

The above approaches can create sketches that closely match an input photo, but it is difficult for their frameworks to generate cartoons in given artistic styles, especially some smooth and fancy styles. Furthermore, in most cases they also do not consider the beautification of stylized output, while our framework can simultaneously handle different styles, portrait-cartoon similarity and visual attractiveness.

General Face Stylization: Meng et al. [33] render artistic paper-cut of human portraits by considering it as an inhomogeneous image binarization problem. Zhao and Zhu [11] use templates to fit a mesh to a face and transfer a set of strokes from a specific painting to a given photograph. Rosin and Lai [34] generate highly abstracted yet recognisable portraits by fitting facial models. However, their objective is not to learn a style of an artist so their scheme cannot be applied to our problem.

III. Data Collection and Analysis

Fig. 2 shows the overview of our optimization approach for data-driven cartoon face synthesis. The input comprise a portrait image, a realistic facial component database $\mathcal{D}_r$ and a cartoon facial component database $\mathcal{D}_c$ of a pre-define style. The output is a cartoon face with the same style of $\mathcal{D}_c$.

A. Database Collection

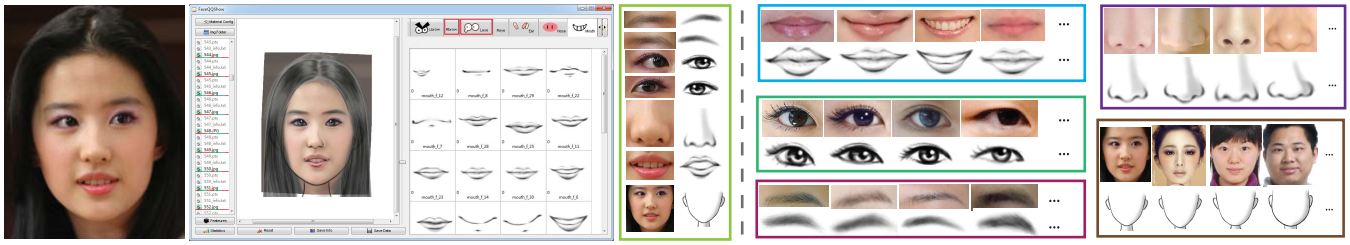
We start the approach with building three databases: one for realistic full faces, one for realistic facial components and one for stylized facial components. The full face database $\mathcal{D}_f$ consists of representative portrait photos downloaded from the Internet and contains 500 frontal or near frontal faces for male and another 500 for female subjects. We select photos to ensure that we cover enough different kinds of facial components for creating various shapes. Next we extract all the facial components from the faces in $\mathcal{D}_f$ to build the database of realistic facial components $\mathcal{D}_r$. We then manually pick representative components of 20 chins, 30 eyebrows, 30 eyes, 16 noses, 30 mouths and 75 types of hair for both male and female photographies from $\mathcal{D}_r$ and use them for building the database of cartoon facial components $\mathcal{D}_c$ by asking artists to draw a stylized version for each representative realistic component. For each real eye we separately draw a stylized version of a single-fold eyelid and a double-fold eyelid. Finally we ask artists to compose cartoons of these faces in $\mathcal{D}_f$, by selecting and composing stylized facial components which are similar to the components of the realistic faces (Fig. 3).

We label every facial component in $\mathcal{D}_r$ with the closest stylized facial component from $\mathcal{D}_c$. Additional styles other than $\mathcal{D}_c$ can be easily added by asking artists to draw stylized facial components in new styles for the elements of $\mathcal{D}_r$. Since there are significant visual differences between different races, e.g., Asian and Caucasian faces, we build databases for different races respectively.

B. Learning the Selection of Components

Previous works all focused on enhancing the attractiveness of real faces [35], [36]. In the contrast, studies on enhancement of facial attractiveness of cartoons have not yet been reported, even though the neural responses related to the evaluation of the attractiveness of cartoon faces have been investigated [37]. Generally, facial components and their composition contribute most for creating stylized faces. A face differs from another not only in terms of different components, but also in its attractiveness which is determined by details and the implicit semantic relationships of all the components. Therefore, it is not meaningful to evaluate the attractiveness of a face just through each individual component. An expedient way to increase attractiveness is to select stylized facial components only according to predefined rules. However, it is not easy to define such rules, because if we consider all the possible combinations, the rules will quickly become intractable to maintain with a growing number of facial components. On the other hand, a restriction to a small subset of possible facial components can avoid awkward synthesis but will result in a limited variety and common artifacts such as creating the same stylized faces for different inputs.

One possibility to encode various relationships and define attractive combinations of facial components is to adopt a data-driven approach based on observational data. Data-driven approaches, including probabilistic graphical models have recently proven to be successful in modeling abstract semantic



(a) (b) (c)

Fig. 3. Database collection. From left to right: (a) input portrait; (b) user interface to label the components of the input portrait; (c) database of labeled components.

relationships, such as component-based shape synthesis [7], outfit synthesis [6], and pattern colorization [38]. Such models automatically extract information and rules from data that have the ability to express some attributes of the input. Since our goal is to combine different stylized facial components in an attractive way and with the necessary variety to represent different input faces, a probabilistic machine learning framework trained by practical data should be appropriate. The higher the probability of a particular component combination in such a network, the higher its attractiveness.

We choose a Bayesian network to learn from a database which consists of stylized faces that are manually created by artists. Bayesian networks are an elegant and efficient method [39] for learning implicit relationships between different components that are consistent with their conditional dependencies. After the training phase our Bayesian network effectively encodes the probability distributions within the space of possible combinations of stylized facial components. An important feature of Bayesian networks is their ability to support inference with incomplete inputs, which is sometimes needed for the selection of facial components. For example, for a certain input face, one may constrain the brows to be of a specific shape and then query the selection of other components in order to form an attractive stylized face according to the trained distribution. Bayesian networks can still return a result under such constrained conditions.

We train separate Bayesian networks for male and female faces as well as for Asian and Caucasian faces. Fig. 4 shows the structure of our Bayesian network. The nodes of the Bayesian networks correspond to the facial regions on which a facial component can be put, the state of each node represents the type of facial component that is added. We build the graph by considering both of the spatial positions and the relative importance of the facial components according to research in psychology [40]. The overall face profile is set as root since it contains all the other components and its shape will directly affect the selection of the other components. The node "Eye" is the parent of node "Brow" because eyes contain more information than brows and they are the parts that catch more attraction at the first glance.

Usually a reasonable quantity of training data is required. For example, 100 plane modes were used to train the networks in [7]. We use the images of our database $\mathcal{D}_f$ as input data and ask three artists to manually create a cartoon face for each real face by selecting stylized facial components from $\mathcal{D}_c$. During the data collection process, we ask the artists to consider not

Nodes	States
Face	face_1, face_2, ... , face_m_F
Eye	eye_1, eye_2, ... , eye_m_E
Brow	brow_1, brow_2, ... , brow_m_B
Mouth	mouth_1, mouth_2, ... , mouth_m_M
Nose	nose_1, nose_2, ... , nose_m_N

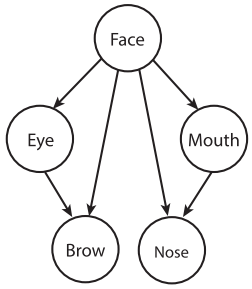


Fig. 4. Structure of the Bayesian network.

only the similarity of the cartoon face to the original input but also a proper relationship between all the components to make the cartoon look more attractive, according to their expertise. For example, we have observed that the artists prefer to put a relatively small mouth on a sharp face profile to make the whole face more attractive. Often they also adjust the position and size of stylized components after combining them, in order to capture and exaggerate the facial characteristics of the input.

We consider each type of stylized component as the state of a random variable. Such a variable has M states if there are M types of its stylized components drawn by artists in this database. Every cartoon face in the database serves as a training sample, and is represented by a five-dimensional vector, $\boldsymbol{x} = \{x_F, x_E, x_B, x_M, x_N\}$, where x_F, x_E, x_B, x_M and x_N represent the face profile, eye, brow, mouth and nose component on a cartoon face respectively. We have 12 states for face contour, eye and mouth as well as 8 states for nose and brow. The graph and the states of random variables are show in Fig. 4.

The Bayesian network corresponds to a joint probability $P(\boldsymbol{X})$ over all the random variables $\boldsymbol{X} = \{X_F, X_E, X_B, X_M, X_N\}$ which is factorized as

$$\begin{aligned} P(\boldsymbol{X}) &= \prod_{s\in\mathcal{S}} P(X_s|\pi(X_s)) \\ &= P(X_F)\cdot P(X_E|X_F)\cdot P(X_B|X_E,X_F) \\ &\quad \cdot P(X_M|X_F)\cdot P(X_N|X_M,X_F), \end{aligned} \tag{1}$$

where $P(X_s|\pi(X_s))$ is a conditional probability distribution (CPD), $\mathcal{S} = \{F, E, B, M, N\}$ is the set of indices for our random variables (i.e., face profile, eye, brow, mouth, and nose), and $\pi(X_s)$ denotes the parent nodes of X_s. All the nodes in the network are discrete and observed, so that the CPDs can be represented as conditional probability tables (CPTs). Our goal is to learn a probabilistic model which is a good representation of the given database. Parameters of the model are the CPTs of the random variables. They determine how accurately the model represents the distribution of the database. We formulate our objective function as

$$\boldsymbol{X} = \underset{\boldsymbol{X}}{\operatorname{argmax}} \prod_{s \in \mathcal{S}} P(X_s|\pi(X_s)).$$

However, even though all the variables are observed, missing dimensions of some samples in the training database will make the model intractable when maximum likelihood estimation is used. For example, a cartoon face without brows (overlapped by hair in original face) may still contain some useful information about the implicit relationships between the remaining components. Therefore, we train the models with the Expectation-Maximization algorithm [41], which is an iterative approach that has the capability to figure out problems involving incomplete data.

C. Learning the Composition of Components

Previous works compose stylized faces according to some pre-defined features [22]. It is difficult for this scheme to express the designer's means of artistic expression in the stylization results. Since the input face and the stylized face are in different domains, a simple linear mapping from the layout of the input face to the layout of the stylized face might make the stylized face unharmonious. Thus, we develop a data-driven approach to address this problem by learning a nonlinear mapping. We adopt the ε-SVR method [8] to let the system learn how to adjust the facial composition of a cartoon in order to improve its visual attractiveness. A facial composition is defined as a feature vector extracted from the facial landmarks. After face alignment, the facial landmarks are aligned according to the two eye centers. Assuming a symmetric face, a coordinate system is defined with its original point as the centroid of two eye centers, the line passing through the eyes is the horizontal axis and the perpendicular line on the nose is the vertical axis. As shown in Fig. 5, a facial composition is represented by a feature vector $\boldsymbol{x} \in \mathbb{R}^{13}$, including coordinates and width of the left brow, coordinates and width of the left eye, y-coordinate and width of the nose, y-coordinate and width of the mouth, as well as y-coordinate and width or the cheek and ordinate of the chin. x_{lbrow} and y_{lbrow} are the coordinates of the left brow center and w_{lbrow} is the width. Similarly, we can define other feature elements in $\boldsymbol{x}$ in the coordinate system.

We ask the artists to compose stylized facial components from $\mathcal{D}_c$ for each face in $\mathcal{D}_f$, by adjusting the position and size of each component to reach a good composition. In this way we obtain a training set $\{(\boldsymbol{x}_1, \boldsymbol{z}_1), ..., (\boldsymbol{x}_l, \boldsymbol{z}_l)\}$, where $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ are feature vectors for each input face and its stylized version, $l = 500$ is the number of training samples. In order to compose a cartoon face automatically during the online synthesis stage, each dimension of its corresponding facial composition needs to be predicted. We learn a ε-SVR for each dimension of $\boldsymbol{z}_i$ by optimizing the following problem:

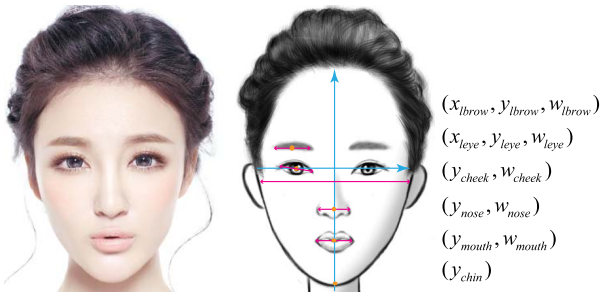


Fig. 5. Feature vector for composition of facial components. Our feature vector contains 13 dimensions and is extracted from both of the real faces and their stylized counterparts, where x and y are coordinates while w stands for width.

$$\begin{aligned} \min_{\boldsymbol{w},b,\xi,\xi^*} \ & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{l}\xi_i + C\sum_{i=1}^{l}\xi_i^* \\ s.t. \quad & \boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b - z_{i,k} \le \varepsilon + \xi_i, \\ & z_{i,k} - \boldsymbol{w}^T\phi(\boldsymbol{x}_i) - b \le \varepsilon + \xi_i^*, \\ & \xi_i, \xi_i^* \ge 0, i = 1, ..., l, \end{aligned} \tag{2}$$

where $\phi(\boldsymbol{x})$ is a mapping function and $C > 0$ is the penalty factor. To solve the problem efficiently, we solve its dual problem instead:

$$\begin{aligned} \min_{\boldsymbol{\alpha},\boldsymbol{\alpha}*} \ & \frac{1}{2}(\boldsymbol{\alpha} - \boldsymbol{\alpha}^*)^T Q(\boldsymbol{\alpha} - \boldsymbol{\alpha}^*) \\ & + \varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} z_{i,k}(\alpha_i - \alpha_i^*) \\ s.t. \quad & \boldsymbol{e}^T(\boldsymbol{\alpha} - \boldsymbol{\alpha}^*) = 0, \\ & 0 \le \alpha_i, \alpha_i^* \le C, i = 1, ..., l, \end{aligned} \tag{3}$$

where $Q_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$ is the kernel matrix, and $K(\cdot,\cdot)$ is the kernel function. We use a radial basis function kernel for this purpose:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)\phi(\boldsymbol{x}_j) = \exp\left\{-\frac{(\boldsymbol{x}_i - \boldsymbol{x}_j)^2}{2\delta^2}\right\}. \tag{4}$$

We leverage the SMO algorithm [8] to optimize the parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}*$, and then predict the position with the input $\boldsymbol{x}_j$ as

$$z_{j,k} = \sum_{i=1}^{l}(-\alpha_i + \alpha_i^*)K(\boldsymbol{x}_i, \boldsymbol{x}_j) + b. \tag{5}$$

There are two hyperparameters C and δ in the model. A grid search method is used to find the best hyperparameters in the range $10^{-10,-8,...,10}$ for C and the range $10^{-5,-4,...,5}$ for δ. And we use 5-fold cross validation strategy and the root mean square error (RMSE) as measure to select the parameters with the minimum regression error. The RMSE is used to measure the difference between the predicted position and the ground truth position by the artists.

IV. Cartoon Face Synthesis

In this section, we describe our cartoon face synthesis method in detail. Starting with an input photo, our method proceeds in five main steps as follows: first, the face image is decomposed into semantically meaningful components (Section IV-A); these components are then matched to potential components in the database with their specific features (Section IV-B); next, the best combination of component candidates is determined through the trained Bayesian network (Section IV-C); the relative positions of the selected cartoon components are further adjusted with the trained ε-SVR (Section IV-D); finally, additional components such hair and glasses are handled for the output (Section IV-E).

A. Facial Image Decomposition

Given a portrait photo, we first compute the face location using the Viola-Jones algorithm for face detection [42]. We then employ a regression-based face alignment method to locate facial landmarks [43]. This method typically detects 88 facial landmarks that are now used to decompose the face into its basic components, including eyes, eyebrows, nose, mouth and the face contour. The regions of eyes, eyebrows and nose are enclosed in bounding boxes defined by their landmarks, while mouth and face contour are directly represented by the polygons defined from their landmarks (22 landmarks for mouth and 21 for face contour). Note that our system only accepts photos with frontal faces by only opening the frontal face detector. As a pre-processing step, a face is first aligned to be horizontal according to the coordinates of eye centres.

B. Facial Component Matching

For each component extracted from the previous step, we search for its K nearest neighbors in the database of realistic facial components $\mathcal{D}_r$. We use $K = 5$ for all of our examples. Since facial components show distinct visual features, we employ different matching schemes for each of the facial components. By matching the components in $\mathcal{D}_r$, we use their corresponding cartoon components in $\mathcal{D}_c$ for the subsequent synthesis process.

Gender Classification: Before the actual matching step, we determine the gender of the face from the input image. A recent survey on gender classification [44] evaluated extensive gender classification methods and found that the C-Support Vector Classification method using face images of resolution 36×36 as input achieved the best overall gender classification rate. Therefore we employ this method to determine the gender and select the corresponding facial component database.

Eyes and Nose: For the eyes, we first scale the eye image to a fixed size (62×100 pixels in all of our examples). The procedure of scaling an eye image is shown in Fig. 6. We first scale and align the two corner points A and B of an eye (located by face alignment) to two points L and R which are two fixed points in a rectangle with a fixed size. The region of the image in the rectangle is the target eye region. We extract nose and eyebrow regions in the same way (described below). We encode each eye region into a

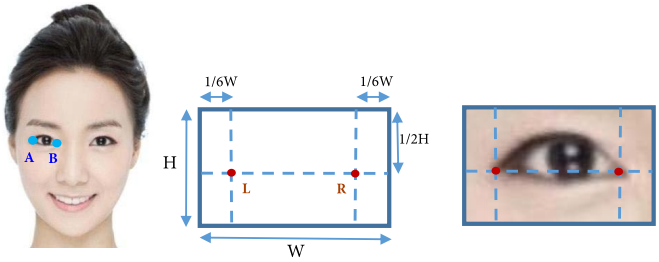


Fig. 6. The procedure for eye region extraction.

2304-dimensional histogram of oriented gradients (HOG) feature vector (see [45]). The HoG features are extracted from gray image. The dimension is determined by the size of window and block. 16×16 is the size for the window, 8×8 for the block, and 8 for the stripe. We select the K nearest neighbors from $\mathcal{D}_r$ by using the Euclidean distance between the feature vectors. For special cases where one or two eyes are closed, we calculate the bounding boxes of facial landmarks of both eyes. An eye is detected to be "closed" if the length-width ratio of its bounding box is less then a user specified threshold (0.2 in practice). For closed eyes, we match them with pre-defined closed eye components in the database. For the nose, we scale the nose region image to a standard size of 70×200 pixels and use the same matching scheme to find the K nearest neighbors in $\mathcal{D}_r$ and their corresponding cartoon components from $\mathcal{D}_c$.

Eyebrows: Eyebrows differ from eyes and nose in that there are usually no clear boundaries between them and the surrounding skin. Therefore, shape-based feature vectors such as HOG are not suitable for measuring the similarity between eyebrow components. We employ the Fourier spectrum distance because of its high matching accuracy for eyebrow recognition [46]. Specifically, we first locate the left and right eyebrows using the facial landmarks, and scale the corresponding images to a fixed size of 66×200. For two eyebrow images of the same size, their Fourier spectrum distance is defined as the χ^2-divergence of their 1D DFT transformed signals [46]. We use this distance measure to find the K nearest neighbors from $\mathcal{D}_r$.

Face Contour and Mouth: Unlike eyes and nose, face contour and mouth can be well represented by their shapes. To compare two sets of face contours, we align their endpoints to the same positions and compute the spatial distance between the other landmark points. The average distance over all the landmark points is used to measure the dissimilarity between the two face contours, with which we find the K nearest neighbors from $\mathcal{D}_r$. We use the same matching scheme for the mouth component.

C. Selection of Facial Components

After obtain K matching components from the database for each component in the input photo, we use them to generate a number of possible combinations for the final output. As mentioned above, we try to generate a stylized face that not only preserves salient characteristics of the original face but also exhibits a high artistic attractiveness. We evaluate the probabilities of all the possible combinations with the

trained Bayesian network from Section III-B and select the combination with the highest probability as the most attractive stylized face. If several combinations share the same highest probability, we randomly choose one of them as the output or show all of them to the users for further selection.

When part of components in a combination are known, the inference process for the trained Bayesian network is achieved by solving the following optimization problem

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}_{\mathcal{Q}} \cup \boldsymbol{X}_{\mathcal{O}} \in \mathcal{X}^{|\mathcal{Q}|+|\mathcal{O}|}}{\operatorname{argmax}} P(\boldsymbol{X}_{\mathcal{Q}}, \boldsymbol{X}_{\mathcal{O}} = \boldsymbol{x}_{\mathcal{O}}),$$

where $\mathcal{Q} \cup \mathcal{O} = \mathcal{S}$. $\mathcal{Q}$ is the set of indices of the unknown variables and $\mathcal{O}$ is the set of indices of the known variables. $\boldsymbol{X}_{\mathcal{Q}}$ is a subset of variables without assignment while $\boldsymbol{x}_{\mathcal{O}}$ is assigned to $\boldsymbol{X}_{\mathcal{O}}$. It is equivalent to the problem

$$\boldsymbol{X}^*_{\mathcal{Q}} = \underset{\boldsymbol{X}_{\mathcal{Q}} \in \mathcal{X}^{|\mathcal{Q}|}}{\operatorname{argmax}} P(\boldsymbol{X}_{\mathcal{Q}} | \boldsymbol{X}_{\mathcal{O}} = \boldsymbol{x}_{\mathcal{O}}),$$

where $\boldsymbol{X}_{\mathcal{Q}}$ is the set of variables to be inferred. In our system, we solve the inference problem by junction tree inference method [47].

D. Composition of Facial Components

After choosing the optimal components via the Bayesian network, we use the corresponding stylized facial components to synthesize the output image. We first build a symmetric coordinate system on the empty canvas and then normalize the detected landmarks by aligning the two eye center points to two fixed points on the coordinate system as shown Fig. 5. For each component, we determine its position, width and length in the coordinate system using its aligned landmarks. A feature vector $\boldsymbol{x} \in \mathbb{R}^{13}$ is then formulated by combining the features of all facial components. Next, we predict a vector $\boldsymbol{z}$ from $\boldsymbol{x}$ with the trained ε-SVR as described in Section III-C, which specifies the relative positions and the sizes of the cartoon components in the output image. Finally, these components are synthesized into a cartoon facial image by combining all the components on the same canvas.

Fig. 7 presents some visual comparisons. The cartoon faces (Fig. 7b) are composed by making their feature vector $\boldsymbol{z}$ as the same as the feature vector of the faces $\boldsymbol{x}$. Although they may look more similar to the original faces compared to these cartoon faces composed by ε-SVR (Fig. 7c), they are less visually attractive. Original faces and cartoon faces are in two different domains. The cartoon face lacks of plenty of texture to represent the depth information, which makes the relative distances between components look larger than the original face even by using the same layout. For example, in the middle column, the distance between the nose and the mouth looks too large for the first, the second and the last faces. Moreover, the distance between the mouth and the chin looks too small for the second face when determining the layout totally according to the original. Actually, it is caused by the pose when taking the photos. In addition, their eyes look too small while ε-SVR tends to make larger cartoon eyes, which appeals to the preference of users. These drawbacks can be improved by learning a model that maps the layout in the original domain to the cartoon domain based on the dataset created by the artists.

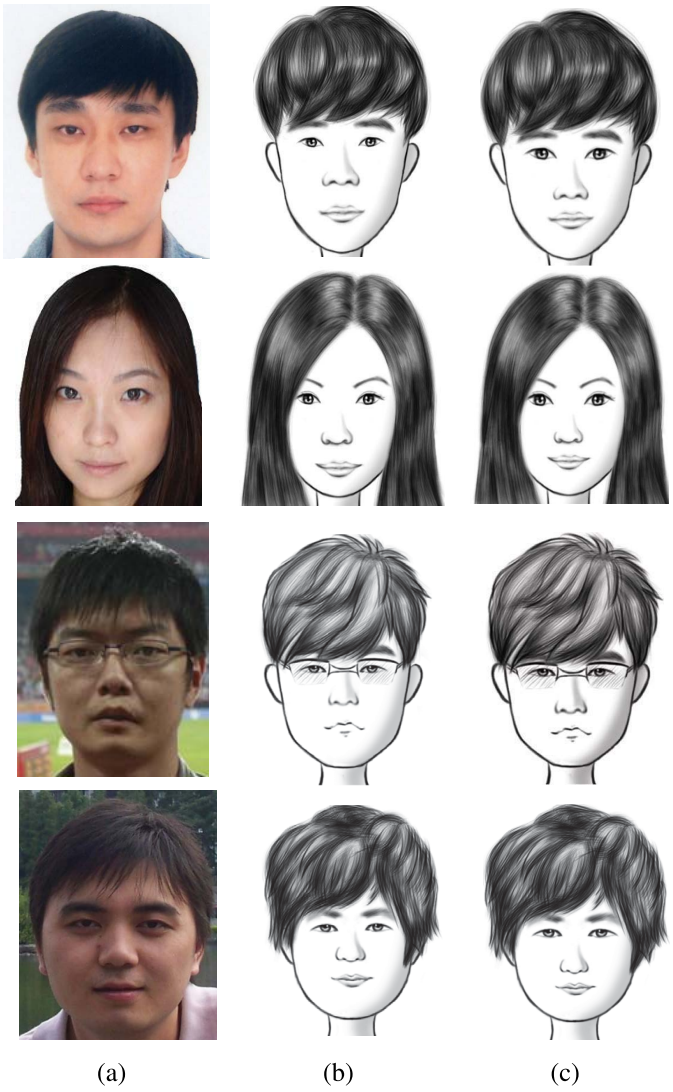


Fig. 7. Comparison of different composition methods. From left to right: (a) input photos; (b) results by direct reference to the photo. (c) results by our ε-SVR model.

E. Handling Additional Components

Some important components such as hair, glasses and eyelid are not covered in previous facial processing steps, but they contribute significantly to the visual appearance of the final stylized output. In this section, we describe specific methods for these additional components.

Hair: Hair is an important component in most portrait images. However, finding a proper cartoon component for a given realistic hair component is a non-trivial task, since both the global shape of the hair and the orientation of the fringe need to be taken into account during the searching process. Traditional shape matching algorithms have shown to be very effective for tasks such as object recognition and retrieval [49], but they cannot be directly used for measuring the subtle differences between different hair contours. Thus, we propose a novel method which combines shape matching and contour alignment to achieve robust hair matching.

Prior to the matching step we extract the hair contour from the input portrait. For doing this we require a background that differs significantly from the hair color. Starting with the

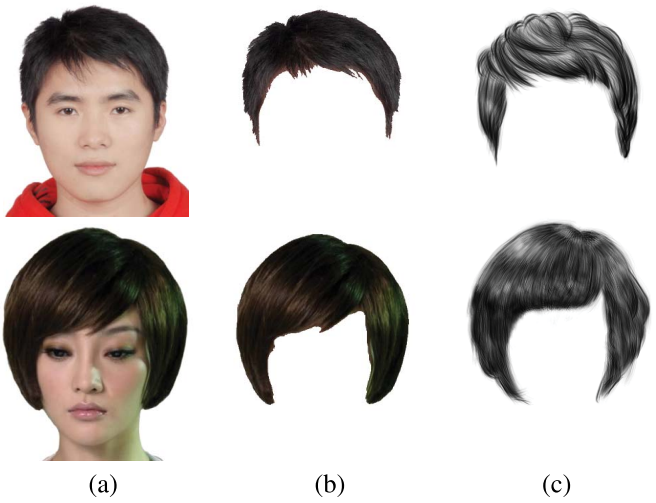


Fig. 8. Hair matting and matching. From left to right: (a) input photos; (b) results of KNN image matting [48]; (c) the best matching hair components in the database.

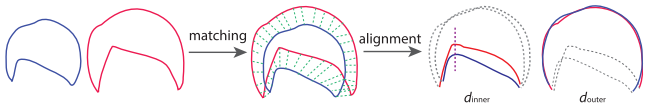


Fig. 9. Procedure of hair matching. Two contours are firstly matched by height functions and then their outer contours are aligned by Horn's quaternion-based method.

detected facial landmarks, we initialize a set of seed points in the hair area and another set of seed points within the background. The hair area is acquired by using the hair detection method in [50]. We treat the area outside the face area and hair area as background. Then the accurate hair region is extracted from the image using the KNN-matting algorithm [48]. The boundary of the hair region is then extracted and smoothed to get the final hair contour. Contours of the hair components in the database are directly extracted from their alpha maps. Fig. 8 presents some examples of extracted hair regions and corresponding cartoon components.

Fig. 9 illustrates the procedure to measure the similarity between two hair contours in a three-step process. We first adopt the height-function based shape matching method as described in [49] to establish a set of corresponding point pairs between the two contours. Then we align the outer contours of the two hair components using Horn's quaternion-based method [51]. By fixing the outer contours, we further align the fringe parts of the two hair components (their inner contours). Practical experience showed us that users are very sensitive about the orientation of their hair within the fringe. Thus, for fringe alignment we only use horizontal translation, such that the orientation of the fringe is not changed during matching. Finally, the distance of the two contours is computed by weighted summing the difference of all point of pairs in the outer contours and that of the inner contours:

$$d = \lambda \cdot d_{\text{inner}} + (1 - \lambda) \cdot d_{\text{outer}} \tag{6}$$

where λ is set to be 0.7 for all of our examples.

Glasses: Glasses appear frequently in portrait photos. We detect the existence of the glasses using a simple but

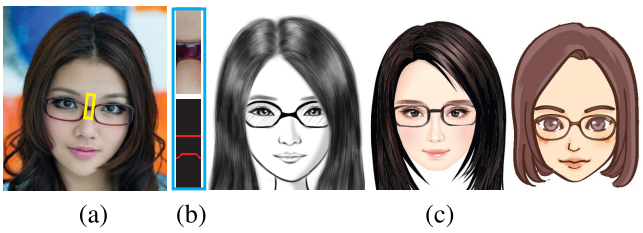


Fig. 10. Detection of glasses. (a) a patch in the central area between the two eyes is extracted; (b) two horizontal edges are detected from the patch (marked in red), which indicates the existence of the glasses; (c) our results of three cartoon styles.

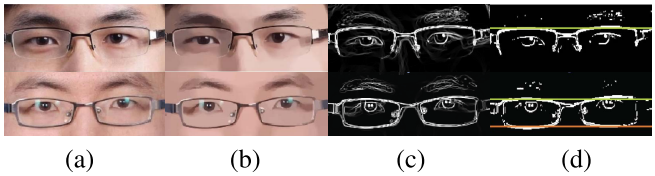


Fig. 11. Detection of frames. (a) input image. (b) L0 gradient smoothing. (c) gradient image. (d) binary image.

effective visual cue, i.e., the frame of the glasses usually forms a horizontal streak between the eyes. Thus we select an image patch in the central area between the eyes and run the Canny edge operator on the patch. A pair of glasses is detected if two horizontal edges are found in this patch. See Fig. 10 for a concrete example.

We determine the frames of the glasses using a similar method. Specifically, we extract a patch covering the two eyes, and detect strong horizontal edges in the region. To handle noise and preserve salient edge structures, we perform an L_0 filter [52] using a patch size of 40×9 before edge detection. If there are two strong horizontal edges enclosing the regions, we use a pair of full-frame cartoon glasses for synthesis; otherwise we use half-frame glasses. Examples with different kinds of glasses are presented in Fig. 11.

Eyelid: The type of the eyelid (double or single) is a distinctive feature for eyes. We use Canny edge detection to get a binary image for the left and right eye. The left eyelid is double if two pulses are detected on the left-top part of its gradient image. To be able to synthesis cartoon faces with this distinctive feature, our cartoon database contains both a single eyelid and a double eyelid for each cartoon eye component.

V. Experimental Results

In this section, we test our method on a variety of real portrait images. We first show that our method can be readily integrated with multiple databases and generate visually convincing facial cartoon images with various artistic styles (Section V-A). We then show through extensive user studies that our method captures salient facial structures and delivers satisfactory visual experience comparable to the manual work by the artists (Section V-B).

A. Visual Examples

Our framework aims to synthesize stylized faces by considering both similarity and attractiveness while existing work pursue to make the stylized face as similar as the original face except for the colors and the textures. We adapted several

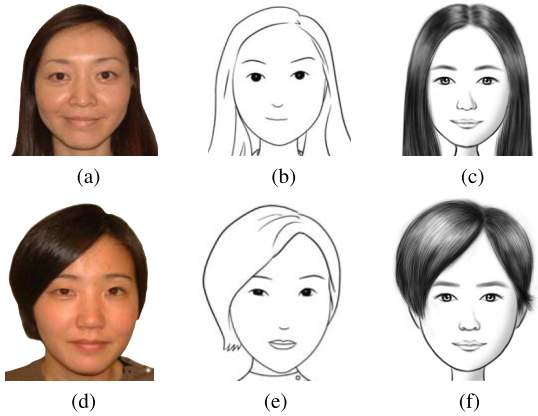


Fig. 12. Comparison with a previous method for sketch synthesis. From left to right: (a)&(d) input photo; (b)/(e) [22]; (c)/(f) our results.

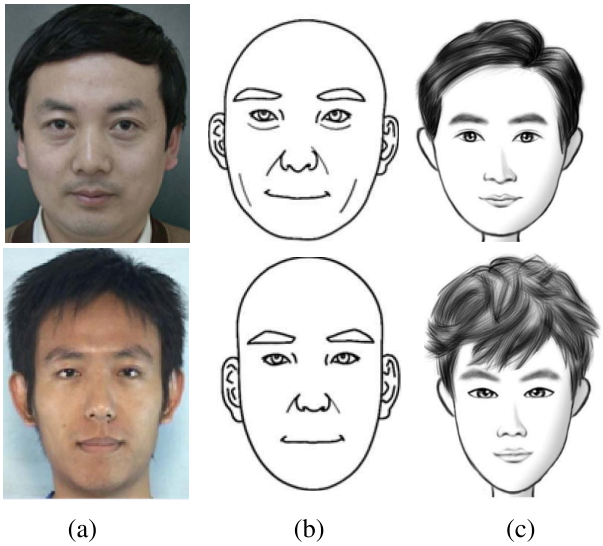


Fig. 13. Comparison with [21]. From left to right: (a) input photos; (b) the results of [21] (c) our results.

photos from existing work and generated stylized faces by our framework for simple comparison, as shown in Figs. 12-14. In Figs. 12-14, we compare with the component-based methods. Those methods synthesize stylized components individually without considering their relationships and directly place them according to the layout of the original face. This scheme will affect the attractiveness of the stylized face in a very different domain. For example, the distance between two eyes looks quite large in Fig. 12(b) and Fig. 12(e) though it is the same as the original face. In addition, one promising advantage of our framework is that it can handle exaggerated styles such as the *FANCY* style which is challenge for existing methods.

We have prepared three sets of stylized facial components for the Asian faces (i.e., *B&W*, *MODERN* and *FANCY*) plus one database for the caucasian faces (the *FASHION* style), and connected them to the real facial components of the database $\mathcal{D}_r$. Any new set of cartoon facial components can be easily incorporated into the current system with minimal manual efforts. In Figs. 1, 17 and 18, we provide synthesis results for various input portrait photos in different styles. Besides, synthesized faces for input faces under different head

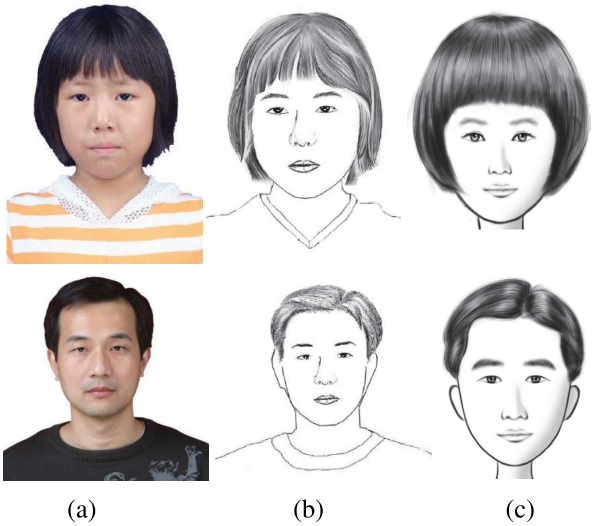


Fig. 14. Comparison with [20]. From left to right: (a) input photos; (b) the results of [20] (c) our results.

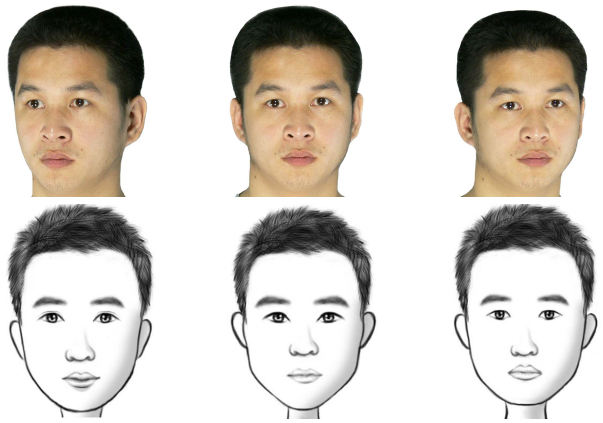


Fig. 15. Comparison using input photos of different poses. Top: input portrait photos; bottom: our results.

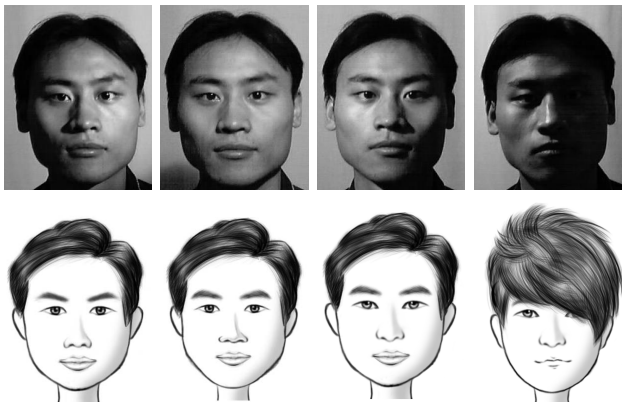


Fig. 16. Comparison using input photos of different illumination.Top: input portrait photos; bottom: our results. Face images are from [53].

poses and different illumination are shown in Figs. 15 and 16, respectively. Our system relies on the accuracy of face alignment result. Face alignment can handle non-frontal head poses and the illumination changes to a certain extent, but might be inaccurate in some extreme cases, such as the rightmost example in Fig. 16. Illumination affects the information

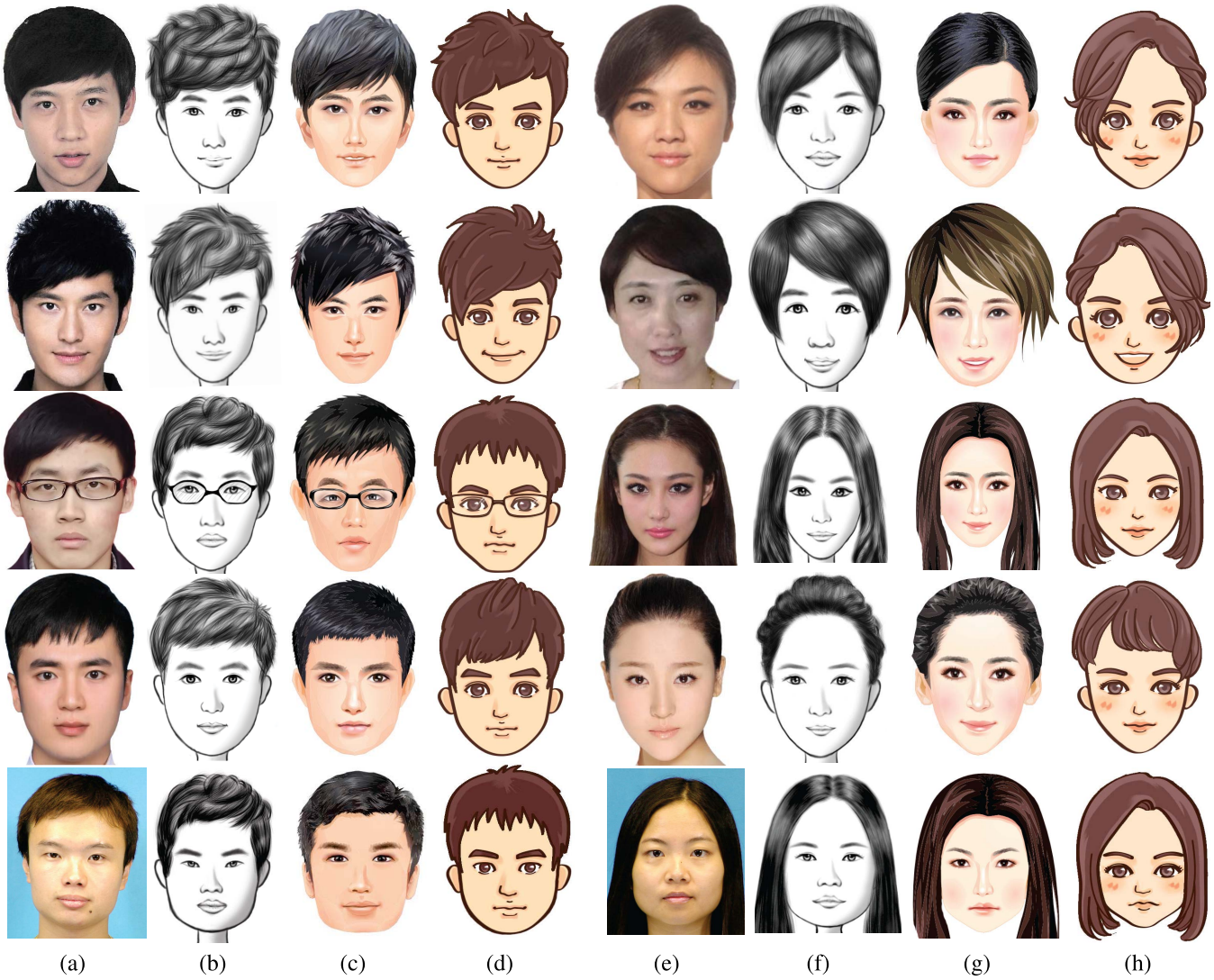


Fig. 17. Results of Asian faces in three styles by our framework. (a)/(e): input photos; (b)/(f): results in the *B&W* style; (c)/(g): results in the *MODERN* style; (d)/(h) results in the *FANCY* style. The two photos at the bottom are from CUHK sketch database [29].

captured in face images and extreme illumination leads to missing textures of facial components, which might affect the accuracy of facial component matching in our system. For components represented by appearance features, the matching process is robust to the illumination changes, since the extraction of HOG features considers the influence of illumination by performing image normalization as preprocessing. For components represented by shape features, the matching process is only affected by the results of the face alignment algorithm.

As can be seen from these visual examples, our method is well suited for practical applications. First, it preserves distinct facial features such as eye size, hair style, face contour shape, etc. and transfers them faithfully to the synthesized results. Second, our method can robustly handle environmental variations in the input portrait images, including cluttered background, illumination change (see Fig. 16) and even varying viewpoints (see Fig. 15, our method generates very stable results for all the input images). Third, our method can also robustly handle the inherent variations in human faces, such as gender, skin colors, hair styles, accessories and races. Specialized modes such as glasses, eyelid or even mustache types can be incrementally added to the system. Finally, our method works reasonably well with different sets of stylized facial components and generate results that are consistent with the original artistic style. Note that previous similarity-oriented patch-based methods cannot generate cartoon faces of MODERN or FANCY styles due to the artistic exaggeration of such styles.

Implementation: We have implemented our system on a PC with 3.4GHz Intel Core i7 and 16GB DDR3 memory. A Bayesian network takes no more than two minutes to train while an ε-SVR model can be learned within 10 minutes. At runtime it takes about 0.8 second on average to synthesize a cartoon face. We have also developed a prototype mobile application on both Android and iOS platforms. An indoor running game that was realized with this application is shown in Fig. 19, which demonstrates that our method can be applied in real-world applications. The main users of this product are young people. Young people especially teenagers often care

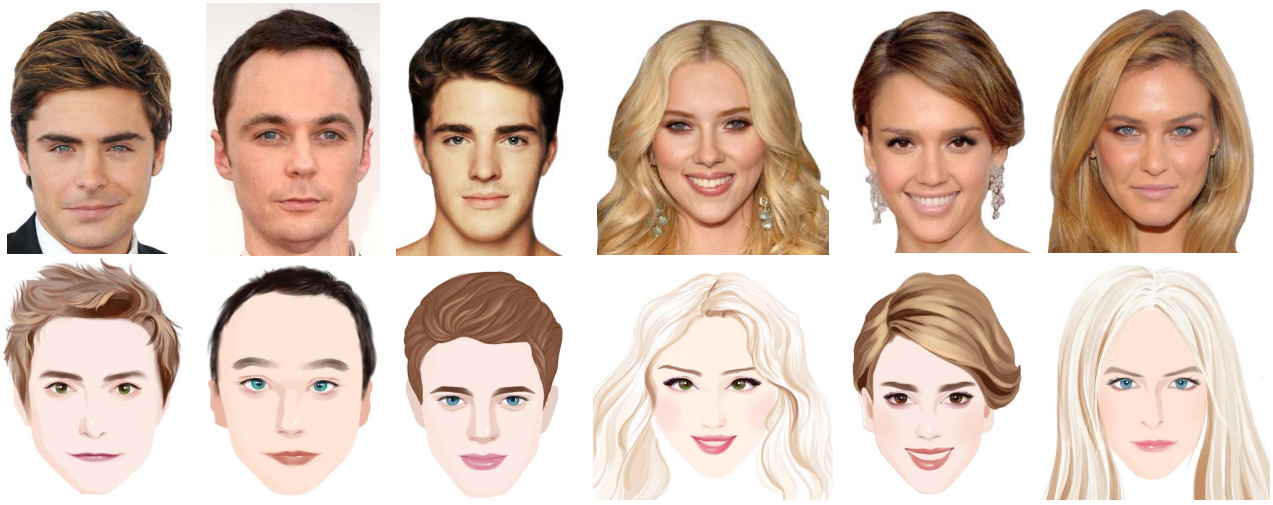


Fig. 18. Results of Caucasian faces in the *FASHION* style. Top: input photos; bottom: our results.

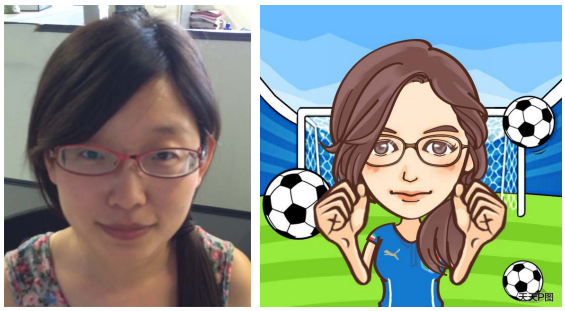


Fig. 19. Cartoon image generated by our mobile software.

about if the resulting cartoons are beautiful, so during development we mainly focus on how to increasing attractiveness instead of maintaining similarity.

B. User Study

In the previous section we have shown that our method can effectively synthesize cartoon facial images with different styles. However, several important aspects about our method remain unclear: Does our method actually preserve the distinctive features of the original face? Does our learning-based framework (Bayesian network + ε-SVR) actually improve over the simple nearest neighbor based method? Can our method compete with an artist? Since the answers to these questions involve human observations and analysis, we perform a set of carefully designed user studies for a quantitative evaluation.

The general settings of our user studies are as follows: 100 portrait images (50 male, 50 female) were collected from the internet. 70 participants (35 male, 35 female) were asked to take the tests. During our user study, each participant would answer a set of questions in sequence. Each question was presented in the form of a single selection from a set of images. The answers of all the participants were collected for quantitative evaluation within each of the user studies. In total, we conducted three user studies to answer the three questions above.

User Study 0: In this user study, our goal is to evaluate the similarity between the input face and the stylized face. We conducted the study not only on the similarity for the whole face but also the similarity for each component. The range of the similarity score is set from 1 to 5 including totally 5 levels. The larger the score is, the more similar the given pair is. For each participant, we randomly selected 50 portrait images (25 male, 25 female) and the corresponding results in the *B&W* style. During the test, each participant saw 50 portraits and their corresponding stylized faces; for each portrait the participant was asked to score the similarity between each stylized component and its input component as well as the similarity between the whole stylized face and its input face. The average scores of each component as well as the whole face are illustrated in Fig. 20a and Fig. 20b. The results include the scores of stylized faces composed by the artists and the stylized faces by our method. For male, the average scores for hair, eye, nose, mouth, brow and profile of our method are 4.23, 4.33, 4.18, 4.37, 3.90 and 4.78 respectively while the average score for the whole face is 3.95. For female, the average scores for hair, eye, nose, mouth, brow and profile of our method are 4.00, 3.85, 4.11, 4.20, 3.37 and 4.81 respectively while the average score for the whole face is 3.63. Most of the similarity scores are larger than 4.0, which demonstrates the effectiveness of the matching procedure in our framework. However, the average scores for stylized faces synthesized by our method are less than 4.0, which is 3.95 for male and 3.63 for female, and the average scores for stylized faces composed by the artists are less than 4.5. One reason is that the stylized faces the *B&W* style lack of enough details such as wrinkles and textures. The enhancement of attractiveness sometimes also affect the similarity since the shape and positions of some facial components may be changed.

User Study 1: In this user study, we try to investigate whether our method is able to capture distinctive facial features during matching of facial components and synthesizing the stylized face. For each participant, we randomly selected a subset of 30 portrait images (15 male, 15 female) and the corresponding results in the *B&W* style. During the test, each participant saw 30 portrait images; for each portrait

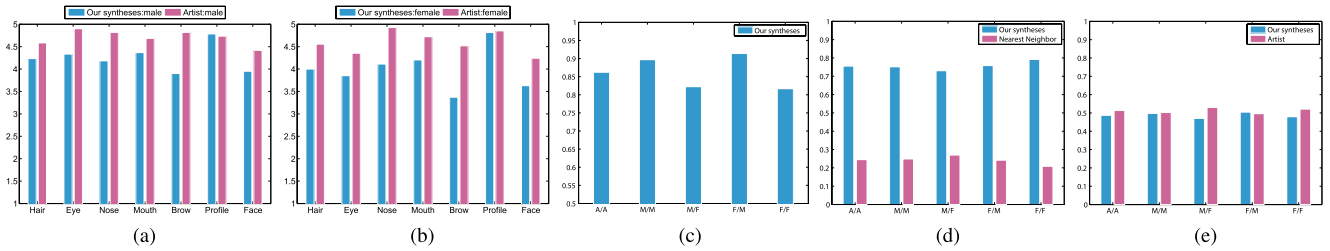


Fig. 20. Results of user studies. From left to right: (a)/(b) Similarity scores of facial components and the whole face in User Study 0. (c) Recognition rates of User Study 1. "A/A": all participants view all results. "M/F": male participants view female results. Similar for "M/M", "F/M" and "F/F". (d) Participants' preference of User Study 2: our results versus Nearest Neighbor. (e) Participants' preference of User Study 3: our results versus artists' drawings.

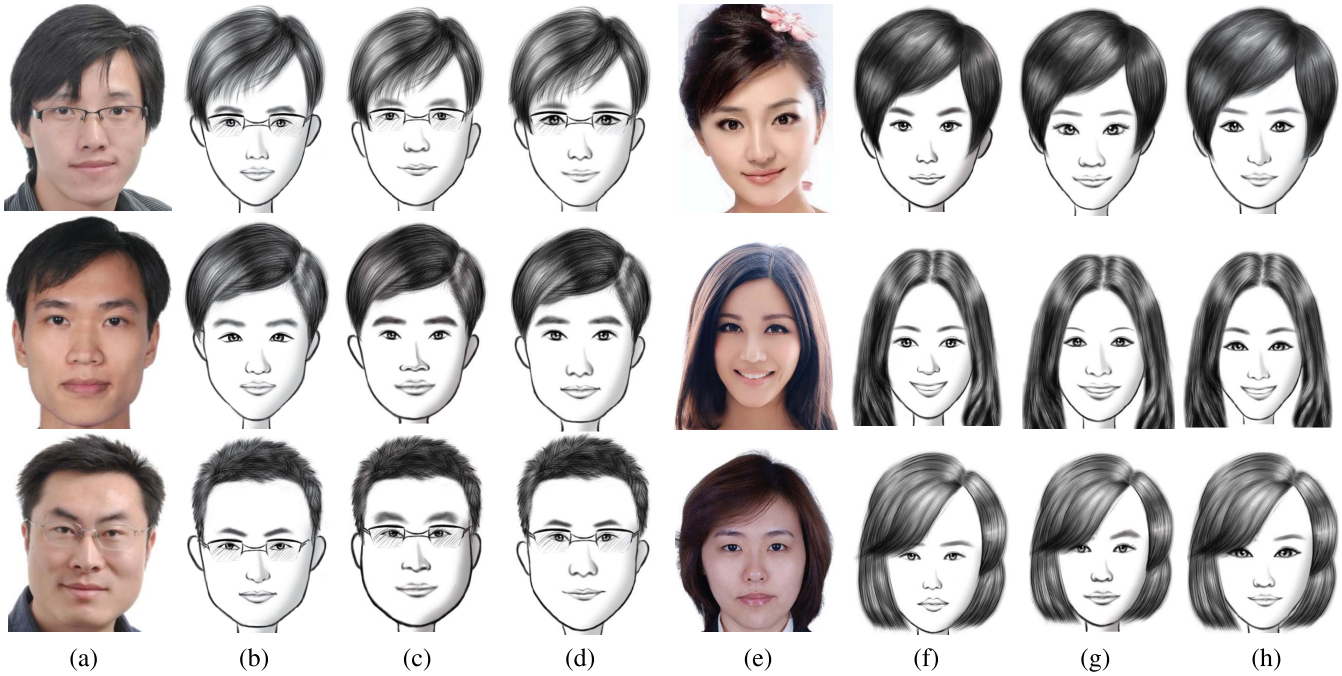


Fig. 21. Cartoon faces in the *B&W* style generated by different approaches. (a)/(e): input photos; (b)/(f): results by manual composition; (c)/(g): results by the baseline method; (d)/(h): results by our method.

we displayed its corresponding stylized result but also three randomly selected cartoons from other input images. The participant was asked to select the cartoon that best describes the current portrait from the four candidates. We collected the answers from the 70 participants and calculated the percentage of the answers that selected the right cartoon for the corresponding portrait (denoted as the recognition rate). Following the user study strategy in [6], we provide five detailed recognition rates (gender of the participants versus the gender of the input portraits). The overall recognition rate (A/A) is 86.2% (See Fig. 20c), which shows that our method is very effective in extracting distinctive facial features of a person and delivering them in a stylized output. The detailed recognition rates for "M/M", "M/F", "F/M" and "F/F" are 89.7%, 82.2%, 91.4% and 81.7% respectively, which reveal some interesting findings with respect to gender differences: male cartoon faces allow a much higher recognition rate than female cartoon faces for all participants (male and female), while male participants seem to be more effective in identifying the subtle differences in female cartoon faces than the female participants.

User Study 2: In this user study, our goal is to evaluate the benefits of the Bayesian network and the subsequent ε-SVR based adjustment. We choose a baseline method, which directly finds the nearest neighbors from the database and uses the corresponding cartoon components for the stylization. During the test, each participant saw 40 portrait images (20 male, 20 female); for each portrait we displayed the synthesized results by our method and the output of the baseline method. The participant was asked to select a cartoon face that best described the given input and at the same time looks attractive. The results are summarized in Fig. 20d. Overall, 75.5% of the participants favored our method. The percentage of participants in favor of our method that includes Bayesian Network and ε-SVR for the gender subsets "M/M", "M/F", "F/M" and "F/F" are 75.2%, 73.0%, 75.8% and 79.2%. This proves that our method significantly outperforms the baseline method in the quality of the stylization results.

User Study 3: Finally, we investigate user preferences between our method and artistic creations using our set of stylized facial components. We selected 40 images and invited artists to select the cartoon components from the database and

compose them to facial images using a commercial platform. The other experimental settings were similar to those of User Study 2. The quantitative results are summarized in Fig. 20e. Overall, 48.5% of the participants selected the result of our method, while 51.5% of the participants favored the artist's work. This slight performance difference seems quite stable with respect to gender differences. Thus, our method can be used as an effective alternative to reduce manual efforts for the creation of facial cartoons.

Fig. 21 shows some cartoon faces generated by the artists, the baseline method and our system. Our method and also the artist's work show clear visual advantages over the baseline method.

Limitations: The main limitation of our system is that we can only deal with frontal or near-frontal faces. The selection of cartoon facial components highly depends on the similarity between the input facial components and the facial components in the realistic facial component database, so a cartoon face generated by our system may not be similar as the original person in the input photo if input face is not frontal. The second limitation is that we currently do not consider some characteristics of input faces, such as wrinkles and special hairstyles, so our cartoon faces also do not contain such details.

VI. Conclusions and Future Work

We have presented a data-driven framework for generating cartoon faces from portrait photographies in a desired style. During the offline analysis stage, our method learns how to select and assemble facial components from databases which are manually prepared by artists. The selection of facial components is learned by a Bayesian network while the composition of components is characterized using an ε-SVR model. During the online synthesis stage, we automatically detect the face in the input photo, select an optimal set of components from the cartoon database via feature matching using a trained Bayesian network, and finally compose them together using the ε-SVR model. We demonstrate that the used Bayesian Network is helpful in arranging selected facial components to an attractive representation of the input. Our experiments show that the system is able to generate face stylizations in a similar quality to what an artist is able to achieve. The system is used in an commercial application for mobile devices.

In the future, we plan to extend our system to synthesize more facial details, such as wrinkles and cheekbones, by adding corresponding components to the database. We are also interested in how to make cartoon faces more lively and animate the stylization results. As a data-driven approach, the effectiveness of our system is mainly limited by the examples in the input databases. Some stylization results may not look very close to the input portraits due to the lack of similar cartoon components in the database. One remedy is to add more components into the database, while another potential solution to apply more advanced algorithms to deform the cartoon components during online synthesis when the matching errors are beyond certain threshold. Lastly, our current system can only handle input portraits in more or less frontal views. How to stylize faces in different views (or even profiles) is another interesting future direction.

Acknowledgements

The authors thank the anonymous reviewers for their constructive comments. They thank Fan Tang for developing the user study system.

References

[1] H. Chen, N.-N. Zheng, L. Liang, Y. Li, Y.-Q. Xu, and H.-Y. Shum, "PicToon: A personalized image-based cartoon system," in *Proc. 10th ACM Int. Conf. Multimedia*, 2002, pp. 171–178.

[2] N. Wang, D. Tao, X. Gao, X. Li, and J. Li, "Transductive face sketch-photo synthesis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 9, pp. 1364–1376, Sep. 2013.

[3] C.-C. Tseng and J.-J. J. Lien, "Colored exaggerative caricature creation using inter- and intra-correlations of feature shapes and positions," *Image Vis. Comput.*, vol. 30, no. 1, pp. 15–25, 2012.

[4] J. Liu, Y. Chen, and W. Gao, "Mapping learning in eigenspace for harmonious caricature generation," in *Proc. 14th Annu. ACM Int. Conf. Multimedia*, 2006, pp. 683–686.

[5] D. I. Perrett, K. A. May, and S. Yoshikawa, "Facial shape and judgements of female attractiveness," *Nature*, vol. 368, no. 6468, pp. 239–242, 1994.

[6] L.-F. Yu, S.-K. Yeung, D. Terzopoulos, and T. F. Chan, "DressUp!: Outfit synthesis through automatic optimization," *ACM Trans. Graph.*, vol. 31, no. 6, pp. 134:1–134:14, 2012.

[7] E. Kalogerakis, S. Chaudhuri, D. Koller, and V. Koltun, "A probabilistic model for component-based shape synthesis," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 55:1–55:11, 2012.

[8] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[9] Y. Zhang, W. Dong, O. Deussen, F. Huang, K. Li, and B.-G. Hu, "Data-driven face cartoon stylization," in *Proc. SIGGRAPH Asia Tech. Briefs (SA)*, New York, NY, USA, 2014, pp. 14:1–14:4.

[10] B. Gooch, E. Reinhard, and A. Gooch, "Human facial illustrations: Creation and psychophysical evaluation," *ACM Trans. Graph.*, vol. 23, no. 1, pp. 27–44, 2004.

[11] M. Zhao and S.-C. Zhu, "Portrait painting using active templates," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Non-Photorealistic Animation Rendering (NPAR)*, 2011, pp. 117–124.

[12] H. Li, G. Liu, and K. N. Ngan, "Guided face cartoon synthesis," *IEEE Trans. Multimedia*, vol. 13, no. 6, pp. 1230–1239, Dec. 2011.

[13] X. Mao, X. Liu, T.-T. Wong, and X. Xu, "Region-based structure line detection for cartoons," *Comput. Vis. Media*, vol. 1, no. 1, pp. 69–78, 2015.

[14] M. Wang *et al.*, "Towards photo watercolorization with artistic verisimilitude," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 10, pp. 1451–1460, Oct. 2014.

[15] K. Qian, B. Wang, and H. Chen, "Automatic flexible face replacement with no auxiliary data," *Comput. Graph.*, vol. 45, pp. 64–74, Dec. 2014.

[16] L. Liang, H. Chen, Y.-Q. Xu, and H.-Y. Shum, "Example-based caricature generation with exaggeration," in *Proc. 10th Pacific Conf. Comput. Graph. Appl.*, 2002, pp. 386–393.

[17] Z. Xu, H. Chen, S.-C. Zhu, and J. Luo, "A hierarchical compositional model for face representation and sketching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 955–969, Jun. 2008.

[18] Y. Liu, Y. Su, Y. Shao, and D. Jia, "A parameterized representation for the cartoon sample space," in *Advances in Multimedia Modeling*, vol. 5916. Berlin, Germany: Springer-Verlag, 2010, pp. 767–772.

[19] S. Wang, D. Zhang, Y. Liang, and Q. Pan, "Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2216–2223.

[20] F. Min, J.-L. Suo, S.-C. Zhu, and N. Sang, "An automatic portrait system based on and-or graph representation," in *Proc. Int. Workshop Energy Minimization Methods Comput. Vis. Pattern Recognit.*, 2007, pp. 184–197.

[21] P. Wei, Y. Liu, N. Zheng, and Y. Yang, "A statistical-structural constraint model for cartoon face wrinkle representation and generation," in *Proc. Asian Conf. Comput. Vis.*, 2009, pp. 466–474.

[22] H. Chen, Z. Liu, C. Rose, Y. Xu, H.-Y. Shum, and D. Salesin, "Example-based composite sketching of human portraits," in *Proc. 3rd Int. Symp. Non-photorealistic Animation Rendering*, 2004, pp. 95–153.

[23] P. Tresset and F. F. Leymarie, "Portrait drawing by Paul the robot," *Comput. Graph.*, vol. 37, no. 5, pp. 348–363, 2013.

[24] I. Berger, A. Shamir, M. Mahler, E. Carter, and J. Hodgins, "Style and abstraction in portrait sketching," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 55:1–55:12, 2013.

[25] J. Xie, A. Hertzmann, W. Li, and H. Winnemöller, "PortraitSketch: Face sketching assistance for novices," in *Proc. 27th Annu. ACM Symp. User Interface Softw. Technol. (UIST)*, 2014, pp. 407–417.

[26] N. Wang, D. Tao, X. Gao, X. Li, and J. Li, "A comprehensive survey to face hallucination," *Int. J. Comput. Vis.*, vol. 106, no. 1, pp. 9–30, 2014.

[27] Q. Liu, X. Tang, H. Jin, H. Lu, and S. Ma, "A nonlinear approach for face sketch synthesis and recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. Jun. 2005, pp. 1005–1010.

[28] X. Gao, N. Wang, D. Tao, and X. Li, "Face sketch–photo synthesis and retrieval using sparse representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 8, pp. 1213–1226, Aug. 2012.

[29] X. Wang and X. Tang, "Face photo-sketch synthesis and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 1955–1967, Nov. 2009.

[30] Y. Song, L. Bao, Q. Yang, and M.-H. Yang, "Real-time exemplar-based face sketch synthesis," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 800–813.

[31] S. Zhang, X. Gao, N. Wang, J. Li, and M. Zhang, "Face sketch synthesis via sparse representation-based greedy search," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2466–2477, Aug. 2015.

[32] S. Zhang, X. Gao, N. Wang, and J. Li, "Robust face sketch style synthesis," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 220–232, Jan. 2016.

[33] M. Meng, M. Zhao, and S.-C. Zhu, "Artistic paper-cut of human portraits," in *Proc. 18th ACM Int. Conf. Multimedia (MM)*, 2010, pp. 931–934.

[34] P. L. Rosin and Y.-K. Lai, "Non-photorealistic rendering of portraits," in *Proc. Workshop Comput. Aesthetics (CAE)*, Aire-la-Ville, Switzerland, 2015, pp. 159–170.

[35] T. Leyvand, D. Cohen-Or, G. Dror, and D. Lischinski, "Data-driven enhancement of facial attractiveness," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 38:1–38:9, Aug. 2008.

[36] A. Khosla, W. A. Bainbridge, A. Torralba, and A. Oliva, "Modifying the memorability of face photographs," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 3200–3207.

[37] Y. Lu, J. Wang, L. Wang, J. Wang, and J. Qin, "Neural responses to cartoon facial attractiveness: An event-related potential study," *Neurosci. Bull.*, vol. 30, no. 3, pp. 441–450, 2014.

[38] S. Lin, D. Ritchie, M. Fisher, and P. Hanrahan, "Probabilistic color-by-numbers: Suggesting pattern colorizations using factor graphs," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 37:1–37:12, 2013.

[39] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[40] I. H. Fraser, G. L. Craig, and D. M. Parker, "Reaction time measures of feature saliency in schematic faces," *Perception*, vol. 19, no. 5, pp. 661–673, 1990.

[41] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc. B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[42] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, 2004.

[43] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," *Int. J. Comput. Vis.*, vol. 107, no. 2, pp. 177–190, 2014.

[44] E. Makinen and R. Raisamo, "Evaluation of gender classification methods with automatically detected and aligned faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 541–547, Mar. 2008.

[45] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. Jun. 2005, pp. 886–893.

[46] Y. Li, H. Li, and Z. Cai, "Human eyebrow recognition in the matching-recognizing framework," *Comput. Vis. Image Understand.*, vol. 117, no. 2, pp. 170–181, 2013.

[47] S. L. Lauritzen and D. J. Spiegelhalter, "Local computations with probabilities on graphical structures and their application to expert systems," *J. Roy. Statist. Soc. B (Methodological)*, vol. 50, no. 2, pp. 157–224, 1988.

[48] Q. Chen, D. Li, and C.-K. Tang, "KNN matting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 869–876.

[49] J. Wang, X. Bai, X. You, W. Liu, and L. J. Latecki, "Shape matching and classification using height functions," *Pattern Recognit. Lett.*, vol. 33, no. 2, pp. 134–143, 2012.

[50] Y. Yacoob and L. S. Davis, "Detection and analysis of hair," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1164–1169, Jul. 2006.

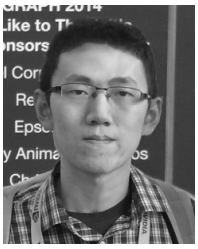
[51] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Proc. Robotics-DL Tentative*, 1992, pp. 586–606.

[52] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via L_0 gradient minimization," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 174:1–174:12, Dec. 2011.

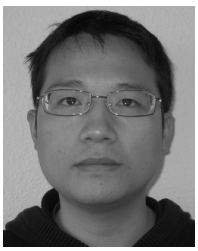
[53] W. Gao *et al.*, "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 1, pp. 149–161, Jan. 2008.

Yong Zhang received the B.Sc. degree in electrical engineering from Hunan University in 2012. He is currently pursuing the Ph.D. degree with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision, machine learning, and probabilistic graphical models.

Weiming Dong (M'11) received the B.Sc. and M.Sc. degrees in 2001 and 2004 from Tsinghua University, China, and the Ph.D. degree from the University of Lorraine, France, in 2007, all in computer science. He is currently an Associate Professor of the Sino-European Laboratory in Computer Science, Automation and Applied Mathematics and National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include image synthesis and image analysis. He is a member of the ACM.



Chongyang Ma received the B.S. degree from the Fundamental Science Class (Mathematics and Physics) of Tsinghua University in 2007 and the Ph.D. degree in computer science from the Institute for Advanced Study of Tsinghua University in 2012. He spent one year as a Post-Doctoral Fellow with the Department of Computer Science, University of British Columbia. He is currently a Post-Doctoral Scholar with the Department of Computer Science, University of Southern California.



Xing Mei (M'11) received the B.Sc. degree in electronic engineering from the University of Science and Technology of China in 2003 and the Ph.D. degree from Chinese Academy of Sciences (CAS) in 2009. He is currently an Assistant Professor of the Sino-European Laboratory in Computer Science, Automation and Applied Mathematics and National Laboratory of Pattern Recognition with Institute of Automation, CASIA. His research interests include image processing, computer vision, and computer graphics. He is a member of the ACM.

Ke Li received the B.Sc. degree in automation from Nankai University and M.Sc. degree in parttern recognition from Shanghai Jiao Tong University, China. He is currently a Senior Engineer of Tencent, Youtu Laboratory. His research interests include face recognition and speaker recognition.

Feiyue Huang is currently the Director of Tencent, Youtu Laboratory. He received the B.Sc. and Ph.D. degrees in computer science from Tsinghua University, China, in 2001 and 2008. His research interests include machine learning and computer vision.

Bao-Gang Hu (M'94–SM'99) received the M.Sc. degree in mechanical engineering from the University of Science and Technology, Beijing, China, in 1983, and the Ph.D. degree in mechanical engineering from McMaster University, Canada, in 1993. From 1994 to 1997, he was a Research Engineer and Senior Research Engineer with C-CORE, Memorial University of Newfoundland, Canada. He is currently a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. From 2000 to 2005, he was the Chinese Director of LIAMA (the Chinese-French Joint Laboratory for Computer Science, Control, and Applied Mathematics). His main research interests include pattern recognition and plant growth modeling.

Oliver Deussen received the degree from the Karlsruhe Institute of Technology in 1996. He was a Post-Doctoral Researcher with the University of Magdeburg on Non- Photorealistic Rendering. In 2000, he was a Professor of Computer Graphics and Media Design by Dresden University of Technology. Since 2003, he has been a Professor of Computer Graphics and Media Informatics, University of Konstanz. He is interested in a number of areas in computer graphics and information visualization. He has authored several books and over 100 refereed publications. He is a member of the ACM Siggraph, Eurographics, and Gesellschaft fuer Informatik. He organized several conferences, was papers Co-Chair of the Eurovis 2004, the Eurogrpahics Symposium of Rendering 2005, the NPAR 2007, the Computational Aesthetics 2009 and papers Co-chair of the Eurographics 2011.