Fast Face Sketch Synthesis via KD-Tree Search

Yuqian Zhang, Nannan Wang(✉), Shengchuan Zhang, Jie Li, and Xinbo Gao

State Key Laboratory of Integrated Services Networks,
Xidian University, Xi'an, China
yuqianz@yeah.net, nnwang@xidian.edu.cn, zsc_2007@163.com,
{leejie,xbgao}@mail.xidian.edu.cn

Abstract. Automatic face sketch synthesis has been widely applied in digital entertainment and law enforcement. Currently, most sketch synthesis algorithms focus on generating face portrait of good quality, but ignoring the time consumption. Existing methods have large time complexity due to dense computation of patch matching in the neighbor selection process. In this paper, we propose a simple yet effective fast face sketch synthesis method based on K dimensional-Tree (KD-Tree). The proposed method employs the idea of divide-and-conquer (*i.e.* piece-wise linear) to learn the complex nonlinear mapping between facial photos and sketches. In the training phase, all the training images are divided into regions and every region is divided into some small patches, then KD-Tree is built up among training photo patches in each region. In the test phase, the test photo is first divided into some patches as the same way in the training phase. KD-Tree search is conducted for K nearest neighbor selection by matching the test photo patches in each region against the constructed KD-Tree of training photo patches in the same region. The KD-Tree process builds index structure which greatly reduces the time consumption for neighbor selection. Compared with synthesis methods using classical greedy search strategy (*i.e.* KNN), the proposed method is much less time consuming but with comparable synthesis performance. Experiments on the public CUHK face sketch (CUFS) database illustrate the effectiveness of the proposed method. In addition, the proposed neighbor selection strategy can be further extended to other synthesis algorithms.

Keywords: Neighbor selection · KD-Tree · Local search · Face sketch synthesis

1 Introduction

Face sketch synthesis has attracted growing attentions due to its applications in both digital entertainment and law enforcement. For example, in mobile Internet application, scene portrait effect generation in video chat makes people enjoy the conversation. In social network, people are more willing to use face sketches generated from their own photos as avatars in Twitter or Facebook. In the cartoon industry, in order to save time, artists use an automatic sketch synthesis system

G. Hua and H. Jégou (Eds.): ECCV 2016 Workshops, Part I, LNCS 9913, pp. 64–77, 2016.
DOI: 10.1007/978-3-319-46604-0_5

to product animations. And in the law enforcement, because a suspect's face photo is not available in most cases, police officers need to match a simulated sketch of a suspect drawn by an artist with the recollection of an eyewitness against a mug-shot database to help identify the suspect. Face sketch synthesis can assist to narrow down the modality gap between face photos and face sketches, and then achieve effective matching. As a result, face sketch synthesis becomes a key technique and research hotspot of sketch-photo recognition.

1.1 Related Work

Currently face sketch synthesis can be mainly categorized as image-based method [7,9] and exemplar-based methods [18,22]. The image-based sketch synthesis methods commonly utilize edges in the input image to produce shadings and strokes. Although image-based methods can produce meaningful stylistic effects in some sense, their results are usually missing important facial details and consequently more like the input photos and rather than the artistic work. On the other hand, exemplar-based methods generate new sketches with rich textures from a set of training face photo-sketch pairs, and can synthesize face sketch of different styles from different training sample sets. Compared with image-based methods, exemplar-based approaches can capture important facial structures more effectively and handle the artistic styles which are difficult to describe in a parametric manner. Thus, exemplar-based methods outperform traditional image-based methods.

There mainly exist two types of exemplar-based face sketch synthesis algorithms: profile sketch synthesis [4,24,27] and shading sketch synthesis [18,23,28]. Compared with profile sketches, shading sketches with both contours and textures are more expressive in representing facial structures and more popular in research. Tang and Wang [18] used a global eigen-transformation to synthesize face sketches from photos. It is not difficult to notice that approximating the sketch drawing process of an artist as a linear process is not accurate. This process is more like a nonlinear process for the complex hair style among different people. To overcome this problem, Liu *et al.* [8] proposed a local geometry preserving based nonlinear method to approximate the mapping function and patch-based reconstruction. Gao *et al.* [6,20] employed sparse representation to adaptively determine the number of nearest neighbors instead of fixed number of neighbors as in [8]. Aforementioned approaches generate local patches independently and cannot learn the face structures well. In order to overcome the drawbacks, a state-of-the-art approach using a multiscale Markov random field (MRF) model [23] has been proposed. This approach can well synthesize complicated face structures and significantly reduce artifacts by learning the face structure across different scales. Nevertheless, the MRF model cannot synthesize new sketches and the optimization problem in solving the MRF is NP-hard. For this problem, Zhou *et al.* proposed a Markov weight fields (MWF) model [28] for face sketch synthesis. The method is capable of synthesizing new sketch patches as a linear combination of candidate sketch patches. There are also some improvements on the MRF model and MWF model, *e.g.* [12–14,21].

In spite of the great achievement that exemplar-based methods can usually produce higher-quality sketches, there exist a common drawback among exemplar-based approaches, that is the matching process to a large amount of training data is too computationally intensive. Patch matching is the most time-consuming part in exemplar-based face sketch synthesis. To solve this problem, Song et al. [16] proposed a real-time spatial sketch denoising (SSD) based sketch synthesis algorithm. This method formulates face sketch synthesis as an image denoising problem. The box filter [19] is applied to the KNN search for achieving real-time performance.

1.2 Our Approach

In this paper, we propose a novel and fast exemplar-based face sketch synthesis method based on KD-Tree. The method adopts the idea of divide-and-conquer (i.e. piece-wise linear) to learn the nonlinear mapping relationship between the facial photos and sketches. Firstly we divides all the training images into overlapping regions and each region is divided into patches, and then KD-Tree is conducted among training photo patches of the same region in the training phase. In the test phase, KD-Tree search are conducted to accelerate the neighbor selection process by matching the test photo patches against the training photo patches. Generally most data shows clustering modality, the efficiency of nearest neighbor selection can be greatly improved by building efficient KD-Tree index structure. In order to enforce local compatibility and smoothness between adjacent synthetic sketch patches, we adopt an average operation for overlapping regions in the final reconstructed results. The effectiveness of the proposed approach is evaluated on the CUHK face sketch (CUFS) database. The experimental results validate that the proposed approach significantly improves the performance of face sketch synthesis in time consumption, and the proposed method is more effective to synthesize sketch compared with the face sketch synthesis method using classical greedy search strategy (i.e. KNN).

2 Face Sketch Synthesis

Due to the large differences between photos and sketches caused by different generation mechanisms and information expression manners, it is difficult to identify the suspect by directly matching the photo-sketch pair. In this condition, by firstly transforming a photo into its corresponding sketch, we can reduce the difference between photos and sketches significantly, then allowing effective matching between the two. In this paper, we assume that the mapping function between photos and sketches is complicated nonlinear. We construct KD-Tree in the training phase and then to search the K nearest neighbor patches of a test photo patch from the KD-Tree in the test phase. The idea of piece-wise linear is applied to learn the nonlinear mapping relationship between photo-sketch pairs. Then we can generate the synthesized sketch patch from the linear combination of K nearest sketches corresponding to the K candidate photo patches. Figure 1 is the whole framework of the proposed method.

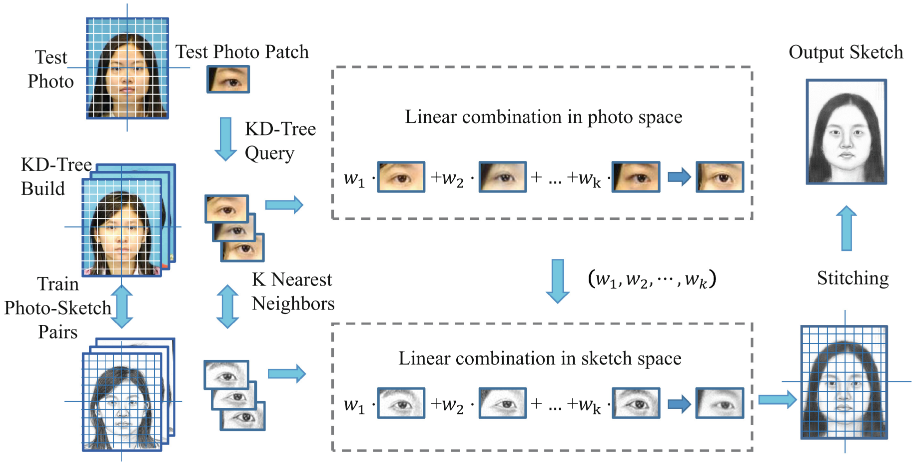


Fig. 1. The framework of our proposed method

KD-Tree is a high-dimensional index tree data structure [26]. KD-Tree has a wide application of approximate nearest neighbor search [1] and nearest neighbor search [3] in large high-dimensional data space, such as K nearest neighbor searching and matching of high-dimensional feature vectors in image retrieval and identification. In practice, most data show a clustering pattern. By constructing effective index structure of KD-Tree, the speed of data retrieval can be greatly improved.

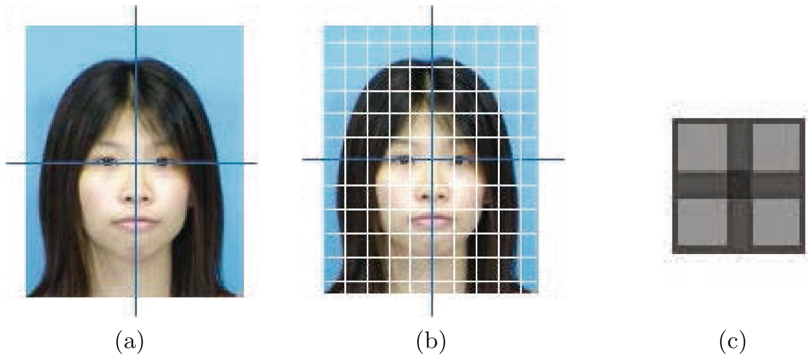


Fig. 2. (a) region division; (b) patch division; (c) overlapping pattern

We assume that all the face images to be studied are plain style and geometrically normalized by fixing the locations of eyes. We adopt the region division strategy to local search. We divide the test photo and all the training photo-sketch pairs into M overlapping regions as shown in Fig. 2(a) for local search.

Compared with global search, local search strategy can not only improve the matching accuracy under the intuition that patches in the same region are more similar to each other, but also reduce the computation in patch matching to save synthesis time. Due to the complexity of face structure, instead of directly learning the global face structure, which might be too complicated to estimate, we target at local patches with simpler structure. Every face region is divided into N overlapping patches in the same way as shown in Fig. 2(b), the overlapping local neighbors can provide global information, sketch patches overlap in the way shown in Fig. 2(c). Let $\mathbf{I}_p^{mn}$ and $\mathbf{I}_s^{mn}$ denote the test photo patch and the sketch patch to be synthesized respectively, $m = 1, 2, \ldots, M$, $n = 1, 2, \ldots, N$. During sketch synthesis, we build KD-Tree for all the photo patches of each region in training photo set as shown in Fig. 3. For a photo patch $\mathbf{I}_p^{mn}$ from test face photo, we find its K most similar photo patches $\mathbf{I}_{pk}^{mn}$ from the training photo set $\mathbf{T}_p^m$ by KD-Tree query, and use its corresponding sketch patches $\mathbf{I}_{sk}^{mn}$ in the training portrait set $\mathbf{T}_s^m$ and reconstruction weights $\mathbf{w}_k^{mn}$ to estimate the sketch patch to be synthesized $\mathbf{I}_s^{mn}$. The underlying assumption is that, if two photo patches are similar, then their sketch patches should also be similar. The proposed face sketch synthesis algorithm is summarized as in Algorithm 1:

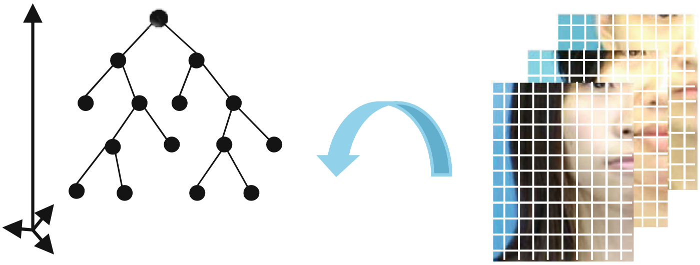


Fig. 3. Constructing a KD-Tree for all photo patches in one region

Stitching sketch patches: In order to strengthen local smoothness and compatibility between adjacent synthetic sketch patches, we adopt an average operation for overlapping areas to make the final synthetic sketch result own the same size as the test photo image.

In step (1) (line 3 in Algorithm 1), when constructing the KD-Tree, we select mean value as threshold of maximum variance dimension to divide the image patch data. We integrate the KD-tree search and best bin first method (BBF) to find the nearest neighbors, and the Euclidean distance is utilized as similarity measure.

In step (2) (line 4 in Algorithm 1), we obtain the weight coefficients by minimize the reconstruction error

$$\varepsilon^{mn}(w) = \|\mathbf{I}_p^{mn} - \sum_{k=1}^{K} \mathbf{w}_k^{mn} \mathbf{I}_{pk}^{mn}\|^2, s.t. \sum_{k=1}^{K} \mathbf{w}_k^{mn} = 1 \qquad (1)$$

Algorithm 1. KD-Tree search based face sketch synthesis

1: **For** $m = 1$ to M
2: **For** $n = 1$ to N
3: Neighbor selection: We build KD-Tree for all the photo patches of each region in training photo set. For a test photo patch $\mathbf{I}_p^{mn}$, we search the K nearest neighbor patches $\mathbf{I}_{pk}^{mn} \in \mathbf{T}_p^m$ at the same region in training photo set by KD-Tree query, $k = 1, 2, \ldots, K$;
4: Computing reconstruction weights: We calculate the reconstruction weights $\mathbf{w}_k^{mn}$ of K photo candidates $\mathbf{I}_{pk}^{mn}$ by minimizing the error of reconstructing $\mathbf{I}_p^{mn}$;
5: Sketch patch estimation: We combine the weight coefficients $\mathbf{w}_k^{mn}$ and the K nearest neighbor sketch patches $\mathbf{I}_{sk}^{mn} \in \mathbf{T}_s^m$ corresponding to the K candidate photo patches $\mathbf{I}_{pk}^{mn}$ linearly to estimate the pseudo-sketch patch $\mathbf{I}_s^{mn}$.
6: **End For**
7: **End For**

This constrained least squares problem can be solved by defining a $K \times K$ matrix Q,

$$Q(i,j) = (\mathbf{I}_p^{mn} - \mathbf{I}_{pi}^{mn})^T (\mathbf{I}_p^{mn} - \mathbf{I}_{pj}^{mn}) \tag{2}$$

and $R = Q^{-1}$, the constrained least squares problem has a close-form solution,

$$\mathbf{w}_k^{mn} = \frac{\sum_{t=1}^{K} R(k,t)}{\sum_{i=1}^{K} \sum_{i=1} R(i,j)} \tag{3}$$

where $k = 1, 2, \ldots, K$. The close-form solution can be directly applied to compute the reconstruction weights.

In step (3) (line 5 in Algorithm 1), the sketch patch to be synthesized can be estimated from the linear combination of the K candidate sketch patches $\mathbf{I}_{sk}^{mn} \in \mathbf{T}_s^m$,

$$\mathbf{I}_s^{mn} = \sum_{k=1}^{K} \mathbf{w}_k^{mn} \mathbf{I}_{sk}^{mn} \tag{4}$$

3 Experiments

In order to verify the effectiveness of the proposed face sketch synthesis algorithm based on KD-Tree, we conduct experiments in the CUFS database, which contains 606 face photo-sketch pairs with photos from three sub databases (*i.e.* the CUHK student database [9], the AR database [10] and the XM2VTS database [11]). The CUHK student consists of 188 photo-sketch pairs where 88 pairs used for training and the rest are for testing. In the AR database which consists of 123 photo-sketch pairs, we randomly choose 80 pairs for training and the rest for test. And for the XM2VTS database, 100 pairs are randomly selected for training and the rest 195 pairs are used for test. Figure 4 shows examples of photo-sketch pairs in the CUFS database. All the face images in the database are in a fontal pose,

Fig. 4. The examples of photo-sketch pairs in CUFS database

with neutral expression, plain style, normal lighting, and without occlusions. In our experiments, the size of face images are aligned to 250 × 200, and the locations of eyes are geometrically normalized by fixing to (75, 125) and (125, 125), respectively. We implement the proposed method using MATLAB and experimented on a computer with i7-4790 Intel core 3.6 GHZ CPU. In following first two experiments, we test the influence of different parameter values to synthesis performance and running time performance. In the third experiment, we evaluate the proposed algorithm against the face sketch synthesis method based on locally linear embedding (LLE) using KNN search [8] (instead of KD-Tree used in our proposed method) and the MRF based approach to validate the efficiency and effectiveness of the proposed method.

In our proposed method, there are following parameters which could affect the final performance: the divided region size, the patch size, the number of neighbors K, the degree of overlapping between adjacent regions and patches. For the patch size, some face details disappear when the size is too large, and noises appear when it is too small. In all the experiments, the patch size is set as 20×20, which can generate good sketch synthesis result. To ensure smooth transition between two adjacent regions or patches, we keep 2/3 region overlapping.

3.1 Synthetic Results of Different Divided Region Size

In the first experiment, we fix the values of all other parameters, and only change the divided region size to see how it impacts on the sketch synthesis result. Figure 5 shows the comparison of region size as 150×150, 100×100, 50×50, and 30×30 with $K = 1$. The sketch synthesis results illustrate that we can obtain a better pseudo-sketch when region size is 50×50. The synthesized sketch is

Table 1. The running time of different divided region size (unit: second)

	Training time	Test time
150 × 150	5.0	32.2
100 × 100	11.6	27.9
50 × 50	13.2	4.8
30 × 30	7.0	3.0

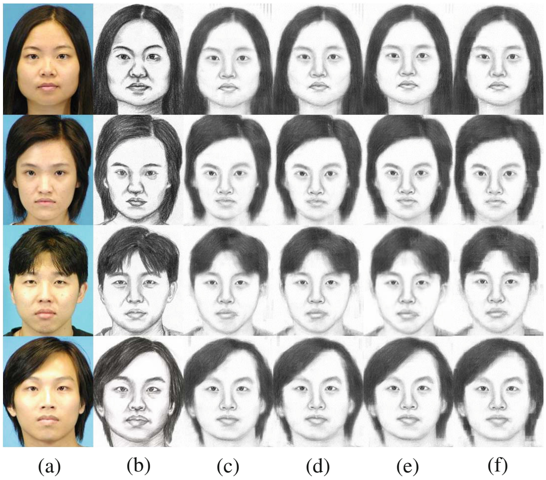


Fig. 5. The face sketch synthesis results of different divided region size: (a) test photo; (b) sketch drawn by the artist; (c) 150×150; (d) 100×100; (e) 50×50; (f) 30×30.

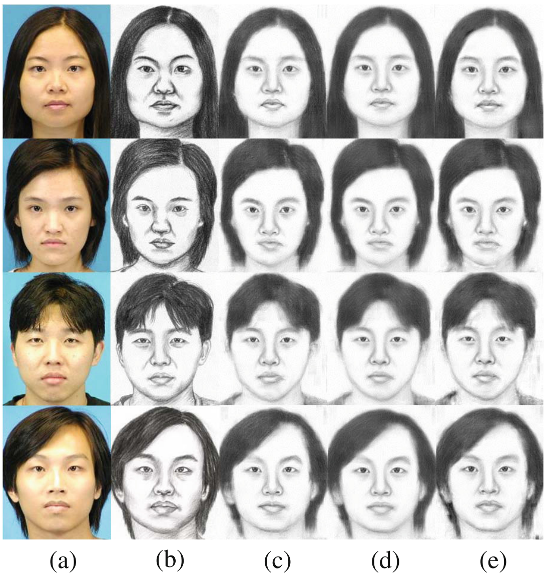


Fig. 6. The face sketch synthesis results of different neighbor number: (a) test photo; (b) sketch drawn by the artist; (c) $K = 1$; (d)$K = 5$; (e) $K = 25$.

noisier when the divided region size is too large (such as 150×150, 100×100), but generated dominant facial structures have no much difference with 50×50. When the region size is too small, there appears mosaic effects as shown in Fig. 5(f).

Table 1 shows the running time when divided region size is set as different values. The training time is the total time of KD-Tree building, and the test time is the synthesis time of each pseudo-sketch generating averaged by the number of test images in a statistical sense. As Table 1 implied, with the divided region size changes, the time consuming of KD-Tree building is irregular, so we can not consider it. On the other hand, the smaller of divided region size, the faster the sketch synthesis rate, and the shorter the time required for KD-Tree query and sketch synthesis. Considering the sketch synthetic quality and time simultaneously, we think that the proposed method achieves better results when region size is 50×50.

3.2 Synthetic Results of Different Neighbor Number

In the second experiment, we fix the values of all other parameters, and only change the number of neighbors K to see how it impacts on the sketch synthesis result. Figure 6 shows the comparison of $K = 1, 5, 25$ with region size 50×50. The sketch synthesis results illustrate that we can obtain a better pseudo-sketch when $K = 5$. We find that when the neighbor number is too small (such as $K = 1$), there are too much noise in synthesized hair, mouth, and chin areas. When K is too large (such as $K = 25$), blurring effect appears in synthesized face contour and the dominant facial details for the linear combination of neighbors. Table 2 shows the running time of pseudo-sketch generating when the number of neighbors K is set as different values. The data in Table 2 show that, with the increase of neighbors, the time consuming of KD-Tree building is still irregular, but the speed of KD-Tree query and sketch synthesis is gradually slowing down, and the method needs more time to synthesize a sketch.

3.3 Results of Different Synthesis Methods

In the third experiment, we compare our method with the LLE method and the MRF method to illustrate the effectiveness and efficiency of the proposed method. For the LLE approach, we set the patch size as 20×20, the number of neighbors $K = 5$, the search region is 5 (that is, search region size $|\triangle| = 30 \times 30$), and keep 2/3 region overlapping. For our method, we set the patch size 20×20, the number of neighbors K=5, the divided region size 50×50, and keep 2/3 region overlapping. Qualitative evaluation results of the proposed method and the LLE method are shown in Fig. 7. The holistic results illustrates that the two method

Table 2. The running time of different neighbor number K (unit: second)

	Training time	Test time
$K = 1$	13.2	4.8
$K = 5$	20.8	9.2
$K = 25$	17.8	157.6

Table 3. Running time of different synthesis methods

	MRF	LLE	Our method
Time (second)	180	670.3	9.2
Program. Language	MATLAB	MATLAB	MATLAB

can both synthesize dominant facial structures and clear contour with less noise. Table 3 shows the running time of different synthesis methods for generating one sketch. The LLE method needs more than 11 min to synthesize a sketch, while our method only cost 9.2 s. The MRF method could achieve the performance of 180 s as reported [23]. All these algorithms could be further speeded up through efficient programming language such as C or C++. We have tested the available MRF codes and it consumes about 8.6 s to synthesize a sketch by using C++ in comparison to 180 s using MATLAB. It is expected that the proposed method could reach the performance of less than 1 s for synthesizing one sketch.

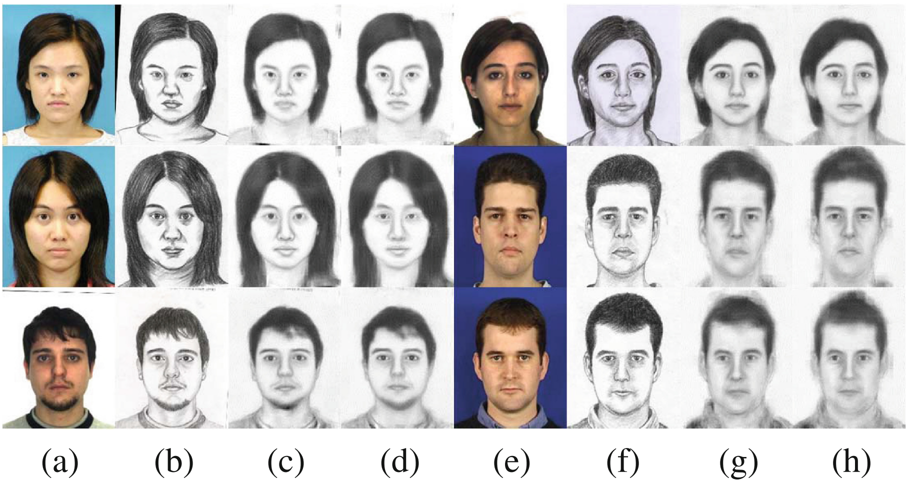


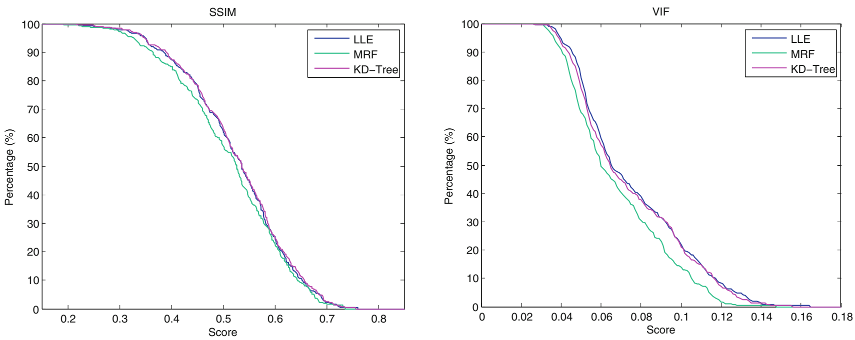
Fig. 7. The face sketch synthesis results of the baseline method and our method: (a) and (e) test photo; (b) and (f) sketch drawn by the artist; (c) and (g) the LLE method; (d) and (h) our method.

In addition, we perform face sketch recognition evaluation on the whole 338 synthesized sketches from the CUFS database. The proposed KD-Tree method is also compared with the LLE method [8] and the state-of-the-art MRF method [23] (synthetic results of the MRF method are generated based on the codes download from [17]) about face sketch recognition accuracy at Rank-1 using two metrics: Fisherface [2] and Null-space linear discriminant analysis (NLDA) [5]. Table 4 shows the face sketch recognition accuracies. We randomly partition 338

Table 4. Rank-1 recognition accuracy using different face sketch synthesis methods and face recognition methods

	Fisherface	NLDA
MRF (%)	78.76	87.72
LLE (%)	84.92	91.04
Our method (%)	84.56	90.12

synthesized sketches into two groups: 150 for training classifiers and the rest 188 for test. We repeat this partition 50 times and the averaged accuracy is reported as shown in Table 4. From the results, we can see that the proposed method and the LLE method achieves higher recognition rates than the MRF approach [23]. The proposed method and the LLE can obtain basically comparable recognition accuracy. The image quality evaluation also conducted on the whole 338 synthesized sketches. Figure 8 shows the quality evaluation results of the proposed method, the LLE method and MRF method using structural similarity index metric (SSIM) [25] and visual information fidelity (VIF) metric [15]. The horizontal axis marks the IQA score and the vertical axis labels the percentage of synthesized sketches whose IQA scores are larger than the score marked on the horizontal axis. Table 5 gives the averaged image quality assessment scores on all 338 synthesized sketches. As shown in Fig. 8 and Table 5, the proposed method and the LLE method have comparable performance, and they are superior compared with the MRF method. In summary, our method could achieve comparable synthesis performance with conventional LLE method but significantly reduce the time consuming.



(a) (b)

Fig. 8. Quality evaluation results of different sketch synthesis methods: (a) SSIM; (b) VIF

Table 5. The quality evaluation scores of different synthesis methods

	SSIM	VIF
The MRF method (%)	51.32	6.78
The LLE method (%)	52.58	7.52
The proposed method (%)	52.83	7.39

4 Conclusions

In this paper, we proposed a fast face sketch synthesis algorithm based on KD-Tree. The algorithm learns the nonlinear mapping function between photos and sketches based on the idea of piece-wise linear. And KD-Tree construction and KD-Tree search are conducted in neighbor selection process to accelerate sketch patch synthesis. The proposed approach is tested in CUFS database including 606 faces. Judging from the results, the proposed method can synthesize face sketch well with smooth and clear facial detail. Compared with the synthesis approaches using KNN, the method can greatly improve the synthetic effectiveness and significantly reduce the synthesis time consumption. In addition, the proposed neighbor selection strategy can be further extended to other synthesis algorithms to speed up the sketch synthesis process.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China (under Grant 61432014, 61501339, and 61671339), in part by the Fundamental Research Funds for the Central Universities (under Grant XJS15049,and JB160104), in part by the China Post-Doctoral Science Foundation under Grant 2015M580818 and Grant 2016T90893 and in part by the Shaanxi Province Post-Doctoral Science Foundation.

References

1. Beis, J., Lowe, D.: Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In: Conference on Computer Vision and Pattern Recognition, pp. 1000–1006 (1997)
2. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 711–720 (1997)
3. Bentley, J.: Multidimensional binary search trees used for associative searching. Commun. ACM **18**(9), 509–517 (1975)
4. Chen, H., Xu, Y., Shum, H., Zheng, N.: Example based facial sketch generation with non-parametric sampling. In: Proceedings of IEEE Conference on Computer Vision, pp. 433–438 (2010)
5. Chen, L., Liao, H., Ko, M.: A new lda-based face recognition system which can solve the small sample size problem. Pattern Recogn. **33**(10), 1713–1726 (2000)
6. Gao, X., Wang, N., Tao, D., Li, X.: Face sketch-photo synthesis and retrieval using sparse representation. IEEE TCSVT **22**, 1213–1226 (2012)

7. Gastal, E., Oliveira, M.: Domain transform for edge-aware image and video processing. ACM Trans. Graph. **30**(4), 1244–1259 (2011)
8. Liu, Q., Tang, X., Jin, H., Lu, H., Ma, S.: A nonlinear approach for face sketch synthesis and recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1005–1010 (2005)
9. Lu, C., Xu, L., Jia, J.: Combining sketch and tone for pencil drawing production. In: Eurographics Association, pp. 65–73 (2012)
10. Martinez, A., Benavente, R.: The AR face database. Technical report 24, CVC, Barcelona, Spain, June 1998
11. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: XM2VTSDB: the extended M2VTS database. In: Proceedings of the International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 72–77, April 1999
12. Peng, C., Gao, X., Wang, N., Li, J.: Superpixel-based face sketch-photo synthesis. IEEE TCSVT **PP**, 1 (2015)
13. Peng, C., Gao, X., Wang, N., Li, J.: Graphical representation for heterogeneous face recognition. IEEE TPAMI (2016)
14. Peng, C., Gao, X., Wang, N., Tao, D., Li, X., Li, J.: Multiple representations-based face sketch-photo synthesis. IEEE TNNLS (2015)
15. Sheikh, H., Bovik, A.: Image information and visual quality. IEEE Trans. Image Process. **15**(2), 430–444 (2006)
16. Song, Y., Bao, L., Yang, Q., Yang, M.: Real-time exemplar-based face sketch synthesis. In: Proceedings of Eureopean Conference on Computer Vision, pp. 800–813 (2014)
17. Song, Y., Bao, L., Yang, Q., Yang, M.: Real-time exemplar-based face sketch synthesis (2014). http://www.cs.cityu.edu.hk/yibisong/eccv14/index.html. Accessed 19 June 2016
18. Tang, X., Wang, X.: Face sketch synthesis and recognition. In: IEEE International Conference on Computer Vision, p. 687 (2003)
19. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of Computer Vision and Pattern Recognition, p. 511 (2001)
20. Wang, N., Li, J., Tao, D., Li, X., Gao, X.: Heterogeneous image transformation. Pattern Recogn. Lett. **34**(1), 77–84 (2013)
21. Wang, N., Tao, D., Gao, X., Li, X., Li, J.: Transductive face sketch-photo synthesis. IEEE TNNLS **24**(9), 1364–1376 (2013)
22. Wang, N., Tao, D., Gao, X., Li, X., Li, J.: A comprehensive survey to face hallucination. IJCV **31**(1), 9–30 (2014)
23. Wang, X., Tang, X.: Face photo-sketch synthesis and recognition. IEEE Trans. Pattern Anal. Mach. Intell. **31**(11), 1955–1967 (2009)
24. Wang, X., Tang, X.: Style and abstraction in portrait sketching. ACM Trans. Graph. **32**(4), 96–96 (2013)
25. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
26. Wikipedia: k-d tree (2016). https://en.wikipedia.org/wiki/K-d_tree. Accessed 19 June 2016

27. Xu, Z., Chen, H., Zhu, S., Luo, J.: A hierarchical compositional model for face representation and sketching. IEEE Trans. Pattern Anal. Mach. Intell. **30**(6), 955–969 (2008)
28. Zhou, H., Kuang, Z., Wong, K.: Markov weight fields for face sketch synthesis. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1091–1097 (2012)