Pacific Graphics (2018)
H. Fu, A. Ghosh, and J. Kopf (Editors)

Short Paper

Bottom-up/Top-down Geometric Object Reconstruction with CNN Classification for Mobile Education

T.Guo#, R. Cui#, X. Qin, Y. Wang*, and Z. Tang

Peking University, Institute of computer science and technology

Abstract

Geometric objects in educational materials are often illustrated as 2D line drawings, which results in the loss of depth information. To alleviate the problem of fully understanding the 3D structure of geometric objects, we propose a novel method to reconstruct the 3D shape of a geometric object illustrated in a line drawing image. In contrast to most existing methods, ours directly take a single line drawing image as input and generate a valid sketch for reconstruction. Given a single input line drawing image, we first classify the geometric object in the image with convolution neural network (CNN). More specifically, we pre-train the model with simulated images to alleviate the problems of data collection and unbalanced distribution among different classes. Then, we generate the sketch of the geometric object with our proposed bottom-up and top-down scheme. Finally, we finish reconstruction by minimizing an objective function of reconstruction error. Extensive experimental results demonstrate that our method performs significantly better in both accuracy and efficiency compared with the existing methods.

Keywords: 3D reconstruction, geometric object classification, sketch generation

1. Introduction

Many educational materials often contain illustrations of geometric objects which are drawn as 2D line drawings. Thus, a well-developed system capable of reconstructing 3D geometric objects from 2D images would have great effects on helping people,especially students, fully understand the geometric structure of these objects [ZWT16b, GWZ*17, ZWT16a]. In this work, we develop a system to recover the depth information of the geometric object from a line drawing image. Specially, our algorithm is efficient enough to run on mobile devices such as smartphones, as shown in Fig.1. With our developed application, a student can select the geometric object from the PDF document as input image. Then our algorithm automatically reconstructs the geometric object in the background and returns the result. The application can display the reconstruction in 3D style from any perspective with the user's interaction, which improves the students' mobile reading and learning experience significantly.

In the past decade, numerous methods to interpret 2D line drawing as 3D object have been proposed. Several early work adopted some geometric regularities such as angularity, face planarity and corner orthogonality [Mar91, SKT01, LL01, LF92]. Later, deduction [Var09, YF11, YF12] or divide-and-conquer strategies [CLT07, ZYL14, XLT12] were used. These methods require the input to

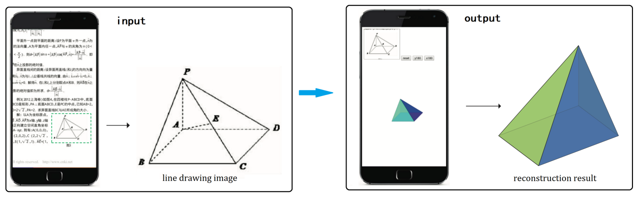

Figure 1: *Illustration the geometric object reconstruction App. The 3D reconstruction result can be showed in any perspective.*

be accurate and complete sketch of the line drawing. To reconstruct the geometric object from the input line drawing image directly, [ZWT16b, ZWT16a] extracted sketches with conventional image processing techniques, then reconstructed the geometric object based on sub-graph matching. Although these methods can get much better results, they still have limitations due to only using hand crafted low-level visual patterns and heuristics rules.

In this paper, we propose a novel approach to reconstruct geometric objects from a single input line drawing image. Given a line drawing image, we first classify the geometric object by CNN. In particular, we use a large set of simulated images combined a few manually-collected images to train the CNN classifier, which improves the classification accuracy. Then, we generate sketch of the geometric object in a bottom-up and top-down scheme based on the classification result. Finally, via the class label and sketch of the geometric object, reconstruction is done by minimizing an objective function of reconstruction error.

# These authors contributed equally to this study and share first authroship.
* Correspondence author. E-mail: wangyongyao@pku.edu.cn

DOI: 10.2312/pg.20181269

2. Proposed Method

The pipeline of reconstructing the 3D shape of the geometric object in a line drawing image is showed in Fig.2. Firstly, we classify the geometric object in the input line drawing image with CNN. Then, we generate a sketch of the geometric object with our proposed bottom-up and top-down scheme. We extract a contour sketch of the geometric object in a bottom-up process, which consists of three steps (Fig. 2(c1-c3)). Top-down strategy is applied to complete the sketch based on the CNN classification result and contour sketch(Fig. 2(c4)). Finally, we obtain the reconstruction result by minimizing an objective function of the reconstruction error.

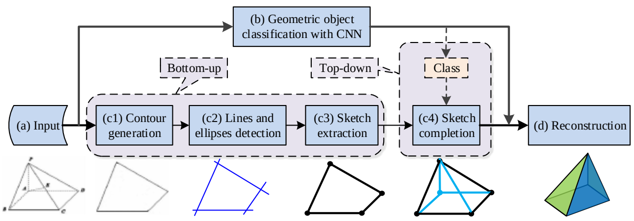

Figure 2: *The pipeline of reconstructing a 3D geometric object.*

2.1. 3D Model Database

Since our method is model-based, we pre-build a 3D model database to aid our reconstruction. The 3D model database contains 24 kinds of 3D models including 17 planar types and 7 curved types such as cuboid and cylinder. It covers almost all the classes of geometric objects that appear in our experimental dataset which is collected from high school geometric educational materials.Each model is controlled by a set of model parameters.

We use $M = (\mathbf{A}, \mathbf{V}, \mathrm{G})$ to represent a 3D model, where $\mathbf{A}$ is the set of model parameters, $\mathbf{V}$ is the matrix consisting of the 3D Euclidean coordinates of the vertices and G is an undirected connected graph denoting which two vertices are connected.

2.2. Geometric Object Classification

The first stage in our algorithm is geometric object classification, which enables us to select a 3D model from the model database that shares the same class with the geometric object illustrated in the input image. We do the classification using a CNN based on the ResNet50 architecture. Training a CNN, however, requires a large amount of data for high accuracy. But data collection is a time-consuming and labor-intensive work. Besides, the distribution among different kinds of classes of the collected data is very unbalanced. In order to alleviate these problems, we train our CNN classification model by using simulated images together with our small manually-collected dataset. We train CNN classifier using two strategies: (a) train a model only on manually-collected dataset; (b) we pre-train the model on simulated images first, then fine-tune the model on the manually-collected dataset. We take the model with the highest classification accuracy for geometric classification.

2.3. Sketch Generation

2.3.1. Bottom-up Process of Contour Sketch Extraction

As shown in Fig.2(c1-c3) , the bottom-up process of contour sketch extraction consists of three steps.

Contour generation. Several pre-processing steps are applied to obtain the contour of the geometric object. First, we convert the input line drawing image to gray-scale, and apply OTSU method [Oht79] to get the foreground image. Next, we search for the connected components of the foreground pixels and select the maximal connected component since the contour of the geometric object is contained in it. Finally, we generate the contour of the geometric object by applying algorithm in [SA85] upon the maximal connected component.

Lines and ellipses detection. With the contour of the geometric object, we apply method in [LWT*13] to detect lines and method in [WHL*15] to detect ellipses (the arcs in the contour can be completed as ellipses during the detection). For lines which are parallel to others within a very small distance are likely to be the different edges of the same line, some of them are removed.

Sketch extraction from the contour. To obtain the contour sketch, we calculate the intersections of the detected lines and ellipses. The intersections correspond to the vertices in the sketch, and the lines and ellipses are converted to the edges in the sketch. We use $S_c = (\mathbf{v}_c, \mathrm{G}_c)$ to represent the contour sketch,where $\mathbf{v}_c$ represents the 2D coordinates of the vertices of the sketch, and G_c is the undirected graph denoting which two vertices are connected.

2.3.2. Top-down Process of Sketch completion

The bottom-up process usually doesn't produce a valid sketch for reconstruction stage. To solve the problem,we utilize a top-down strategy to complete the sketch.

Problem definition. Based on the class (Cla) and contour sketch (S_c) of the geometric object, we complete the sketch in a top-down manner. It can be defined as a problem to maximize a posteriori probability $P(S|Cla, S_c) \propto P(Cla, S_c|S)P(S)$, where S is the sketch we aim to obtain.

Our goal of sketch completion contains two aspects: (a) complete the undirected connected graph(i.e., G_S) of S as perfect as possible; (b) find out the 2D coordinates of the vertices(i.e., $\mathbf{v}_S$) of S. The information G_m provided by selected 3D model $M = (\mathbf{A}_m, \mathbf{V}_m, \mathrm{G}_m)$ and information provided by contour sketch $S_c = (\mathbf{v}_{S_c}, \mathrm{G}_{S_c})$ is helpful for sketch completion. In this sense, the posteriori probability can be converted to $P(S|\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c})$ and the maximum problem can be defined as following:

$$\max_S P(\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c}|S)P(S). \quad (1)$$

Let $E(\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c}|S) = -logP(\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c}|S), E(P(S)) = -log(P(S))$. Then maximizing (1) is equivalent to:

$$\min_S \Big(E(\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c}|S) + E(P(S)) \Big). \quad (2)$$

Corresponding with the goal of sketch completion, the first term $E(\mathrm{G}_m, \mathrm{G}_{S_c}, \mathbf{v}_{S_c}|S)$ in (2) is designed according to the following two principles: (a) G_S should contain G_{S_c} and be as close as possible to G_m; (b) $\mathbf{v}_S$ should match with the 2D coordinates in the input line drawing image as much as possible. To the second term $E(P(S))$ in (2), we observe that once the number of vertices of a sketch is determined, the more vertices whose degree is larger than 2 the sketch contains, the more likely the sketch corresponds to a geometric object. So it is reasonable to define $E(P(S))$ by the number of vertices whose degree is not larger than 2 in the sketch. Considering all the above together, the sketch completion process is converted to:

$$\min_{\{G_S,\mathbf{v}_S\}}\Bigg(\lambda_v\sum_{k=1}^{n_{\mathcal{V}}}\left\|(v_{S,k}-v_{S_c,k})\right\|^2+\lambda_v\sum_{j=1}^{n_{\mathcal{L}}}\left|(\theta_{l_j}-\theta_{l_{j'}})\right| + \lambda_G dis(G_S,G_m)+\lambda_G dis(G_{S_0},G_{S_c}) + \lambda_p\sum_{i=1}^{n_{\mathcal{V}_S}}I(v_i)+\lambda_d\sum_{i=1}^{n_{\mathcal{V}_S}}I(d_i\le 2)\Bigg), \tag{3}$$

where λ_G, λ_v , λ_p and λ_d are the corresponding weights for each term, $n_{\mathcal{V}}$ is the number of vertices in the contour sketch, l_j and $l_{j'}$ are the jth pair of edges that are expected to be parallel in 3D geometric object, θ is the angle of the line in the image line drawing which corresponds to an edge in the sketch, $n_{\mathcal{L}}$ is the number of parallel lines pair in the 3D geometric object, $n_{\mathcal{V}_S}$ is the number of vertices in S, and I is the indicator function whose value is set to 1 when the 2D coordinate of v_i is determined, or otherwise is set to 0. d_i is the degree of the ith vertex v_i.

Solution to (3). Finding the optimal solution to (3) is not a trivial problem, but we can simplify it by initializing S with S_c, therefore setting the second and the third term in (3) to zero. Then, we iteratively update S by completing G_S and finding out the coordinates of the newly added vertices in G_S based on the parallel projection constraint, as shown in Algorithm 1.

Algorithm 1 Sketch completion

Initialization: initialize S with S_c, $S^0 = S_c$, (i.e., $G_S^0 = G_{S_c}$, $\mathbf{v}_S^0 = \mathbf{v}_{S_c}$); $t \leftarrow 0$.
Update $\mathbf{G}_S^t$ with the edges that should be connected to the existing vertices, $t \leftarrow t+1$.
while $|value_{(3)}^t - value_{(3)}^{t-1}| > \varepsilon$ **do**
  for each ($v_h \in \mathbf{G}_m$ && $v_h \notin \mathbf{G}_S$) **do**
    Add v_h and the corresponding edges to G_S^t to narrow $dis(G_S, G_m)$;
    Use parallel constraint to update $\mathbf{v}_S^t$;
    Calculate the value of (3);
  end for
  Select the S with the smallest value of (3) in the last for loop to update S^t, $t \leftarrow t+1$.
end while
Return S^t.

2.4. Reconstruction

Reconstruction is the process of finding an instance of the selectded model by minimizing the reconstruction error of an objective function. We define the objective function as the coordinate residuals of the matched vertices of sketch and the selected model as following:

$$f=\sum_{k=1}^{n_C}\left\|K(RV_{i_k}+t)-v_{j_k}\right\|^2, \tag{4}$$

where n_C is the number of the matched vertices pairs, $K = \begin{pmatrix}100\\010\end{pmatrix}$ is the parallel projection matrix, R is the rotation matrix, t is the translation vector, and V_{i_k}, v_{j_k} are the corresponding coordinates of the matched vertices, V_{i_k} from 3D model and v_{j_k} from sketch.

According to an orthogonal constraint of R, we convert the minimizing reconstruction error to an optimal problem as following:

$$\tilde{\mathbf{A}},\tilde{R},\tilde{t}=\arg\min\sum_{k=1}^{n_C}\left\|K(RV_{i_k}+t)-v_{j_k}\right\|^2, \tag{5}$$

where $\tilde{\mathbf{A}}$ is the optimal vector of model parameters, $\tilde{R}$ is the optimal rotation matrix, $\tilde{t}$ is the optimal translation vector, and $R^T R = I$.

With the classification result, we select the model from the 3D model database and apply VF2 algorithm [CFSV04] to find the correspondence between vertices of sketch and selected model. Based on the correspondence, we take 2D coordinates of the vertices of the sketch and 3D coordinates of the vertices of the selected model to the objective function. By adopting an alternative minimization algorithm in [XLT12] , we obtain the final optimal values of $\mathbf{A}$, R, and t to generate an instance of the selected model in 3D style.

3. Experiments

We build two kinds of datasets for our experiments as described as follows: The manually-collected dataset which is photographed from high school geometry studying materials, contains 4858 line drawing images, including 24 classes of geometric objects, 17 planar types and 7 curved types. We split this dataset into 4018 images as training set and 840 images as testing set. To overcome the lack of manually collected images and unbalanced distribution among different classes, we generate the simulated dataset with 12000 simulated line drawing, 500 images for each class. Some examples of reconstruction results are shown in Fig.3.

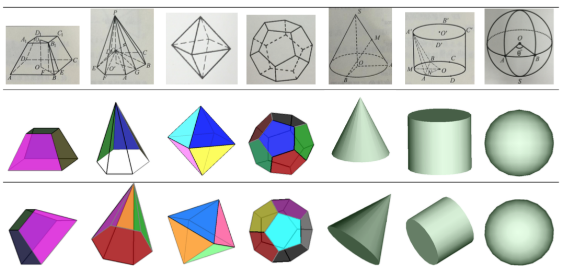

Figure 3: *Reconstruction results of some planar and curved geometric objects.*

3.1. Methods Comparison

We compare with the most related methods: SGOR [ZWT16b], CGOR [ZWT16a] and GO3R [GWZ*17], and we utilize the same accuracy metric with them, defined as: $f_a = |\mathbf{F}_{successful}|/|\mathbf{F}|$, where $\mathbf{F}_{successful}$ is successfully reconstructed (correct classification, correct contour sketch and small reconstruction residual) geometric object set, and $\mathbf{F}$ is the experimental dataset.

Result and analysis. As shown in Table 1, the accuracy of our method is higher than the others significantly. Since SGOR and CGOR only use bottom-up process, inaccurate and incomplete sketch increases the risk of failed reconstruction. Besides, they use sub-graph isomorphism method to match the sketch with all models in the database and select the best one, which means that multiple models may correspond to a same sub-graph but the select model might be wrong. These two limitations mainly cause the low reconstruction accuracy and high time consuming. Our method, with CNN classifier and sketch generation in a bottom-up and top-down scheme, significantly improves the accuracy of geometric objects reconstruction.

Table 1: *Reconstruction accuracy and efficiency comparation.*

Method	Accuracy	Speed (s/image)
SGOR	72.9%	19.881
CGOR	81.2%	–
GO3R	85.2%	0.586
Ours	**93.0%**	**0.467**

3.2. Geometric Objects Classification Comparison

In this section, we compare the accuracy of geometric objects classification of our method with GO3R. We only compare classification accuracy with GO3R because the other related methods (i.e., SOGR and CGOR) do not utilize geometric objects classification before reconstruction. We also compare the accuracy of two training strategies for our method. The base net for geometric object classification is ResNet50. Models are evaluated on the manually-collected testing set. Two strategies are utilized to train the model: (1) only on manually-collected training set; (2) pre-train on simulated dataset and fine-tune on manually-collected training set.

Table 2: *Mean classification accuracy comparison.Our-1 and our-2 denotes donates two different training strategies*

Method	Mean classification accuracy
GO3R	87.62%
ours-1	92.91%
ours-2	**94.71%**

Result and analysis. From Table 2 , we observe a) that our method based on CNN with different training strategies both perform much better than GO3R, b) the model with pre-train and fine-tune strategy performs better than the model only train on manually-collected training set. Since GO3R use hand crafted low-level visual patterns and heuristics rules for classification, which will get wrong result when the input line drawing image is in poor quality or the geometric object is relatively complicated. Besides, the pre-train and fine-tune strategy alleviates the problem of data collection and unbalanced distribution among the classes, improving the accuracy much more.

4. Conclusion and Future Work

An effective and efficient 3D reconstruction approach is proposed for geometric objects from single line drawing images. The main contributions include: (a) simulated images are used together with the manually collected training data to train the CNN for geometric objection classification, alleviating the problem of dataset collection and unbalanced distribution among different classes. (b) a bottom-up and top-down scheme is proposed for sketch generation, making full use of the classification result produced by CNN. Extensive experimental results demonstrate the proposed method can achieve much better performance than the existing methods in both accuracy and efficiency. In the future, we plan to explore the features of geometric objects learned by our CNN classification network and utilize them for sketch extraction.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China under Grant 61673029 and The National Key Research and Development Program of China under Grants 2016YFB1000603. This work is also a research achievement of Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

References

[CFSV04] CORDELLA L. P., FOGGIA P., SANSONE C., VENTO M.: A (sub)graph isomorphism algorithm for matching large graphs. *TPAMI 26*, 10 (2004), 1367–1372. 3

[CLT07] CHEN Y., LIU J., TANG X.: A divide-and-conquer approach to 3d object reconstruction from line drawings. 1–8. 1

[GWZ*17] GUO T., WANG Y., ZHOU Y., HE Z., TANG Z.: Geometric object 3d reconstruction from single line drawing image with bottom-up and top-down classification and sketch generation. In *ICDAR* (2017), pp. 670–676. 1, 3

[LF92] LECLERC Y. G., FISCHLER M. A.: An optimization-based approach to the interpretation of single line drawings as 3d wire frames. *IJCV 9*, 2 (1992), 113–136. 1

[LL01] LIU J., LEE Y. T.: Graph based method for face identification from a single 2d line drawing. *TPAMI 23*, 10 (2001), 1106–1119. 1

[LWT*13] LIU D., WANG Y., TANG Z., LI L., GAO L.: Automatic comic page image understanding based on edge segment analysis. In *IST/SPIE Electronic Imaging* (2013), p. 90210J. 2

[Mar91] MARILL T.: Emulating the human interpretation of line-drawings as three-dimensional objects. *IJCV 6*, 2 (1991), 147–161. 1

[Oht79] OHTSU N.: A threshold selection method from gray-level histograms. *IEEE Trans. Syst., Man Cybern. 9*, 1 (1979), 62–66. 2

[SA85] SUZUKI S., ABE K.: Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing 30*, 1 (1985), 32–46. 2

[SKT01] SHOJI K., KATO K., TOYAMA F.: 3-d interpretation of single line drawings based on entropy minimization principle. 90. 1

[Var09] VARLEY P. A. C.: Comments on "what the back of the object looks like: 3d reconstruction from line drawings without hidden lines. *TPAMI 31*, 8 (2009), 1535–1536. 1

[WHL*15] WANG Y., HE Z., LIU X., TANG Z., LI L.: A fast and robust ellipse detector based on top-down least-square fitting. In *BMVC* (2015), pp. 156.1–156.12. 2

[XLT12] XUE T., LIU J., TANG X.: Example-based 3d object reconstruction from line drawings. In *CVPR* (2012), pp. 302–309. 1, 3

[YF11] YONG T. L., FANG F.: 3d reconstruction of polyhedral objects from single parallel projections using cubic corner. 1025–1034. 1

[YF12] YONG T. L., FANG F.: A new hybrid method for 3d object recovery from 2d drawings and its validation against the cubic corner method and the optimisation-based method. *Computer-Aided Design 44*, 11 (2012), 1090–1102. 1

[ZWT16a] ZHENG J., WANG Y., TANG Z.: Context-aware geometric object reconstruction for mobile education. In *MM* (2016), pp. 367–371. 1, 3

[ZWT16b] ZHENG J., WANG Y., TANG Z.: Recovering solid geometric object from single line drawing image. *Multimedia Tools and Applications*, 17 (2016), 1–22. 1, 3

[ZYL14] ZOU C., YANG H., LIU J.: Separation of line drawings based on split faces for 3d object reconstruction. In *CVPR* (2014), pp. 692–699. 1