Learning Report on Co-segmentation of 3D Shapes

Yingyan Wu, CSSE, SZU

**Abstract**

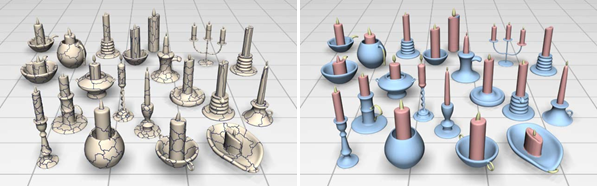
In recent years, co-segmentation of a set of 3D shapes, i.e., segmentation of the shapes as a whole into consistent semantic parts with correspondences, has received increased attention. Many scholars have made corresponding research on the co-segmentation of 3D shapes. This report is mainly about the techniques of co-segmentation of 3D shapes based unsupervised learning. Ruizhen Hu et al.[1] present a novel algorithm about subspace clustering. Sidi et al.[2] focus on the method via descriptor-space spectral clustering. And Meng et al.’s [3] approach is about iterative multi-label optimization. Starting from the over-segmentations of the shapes, [1] generates the segmentation by grouping the primitive patches of the shapes directly and obtains their correspondences simultaneously.[3] clusters the primitive patches to generate an initial guess. Then, it iteratively builds a statistical model to describe each cluster of parts from the previous estimation, and employs the multi-label optimization to improve the co-segmentation results. [2] shows that they can handle the challenging input sets, which may exhibit significant shape variability where the shapes do not admit proper spatial alignment and the corresponding parts in any pair of shapes may be geometrically dissimilar. In summary. the experimental results show that the their algorithm jointly extracts consistent parts across the collection in a better manner compared to other techniques, but they all have certain limitations at the same time.

**Keywords:** Co-segmentation; Unsupervised Learning; Descriptor-Space Spectral Clustering; Subspace Clustering; Iterative Multi-Label Optimization;

# Introduction

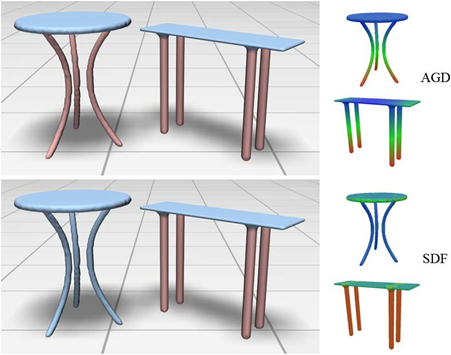
Segmentation of a 3D shape into semantic parts is a fundamental task in high-level shape analysis and processing[4,5,6]. The increasing research trend towards the high-level analysis of 3D shapes to derive structural and semantic shape information. This research heavily relies on the semantic knowledge based shape segmentation . It has been demonstrated that more knowledge can be inferred from multiple shapes rather than an individual shape and co-analysis of the set produces better segmentations than the single-shape algorithms[2,6,7,8]. The existing co-segmentation algorithms can be classified into two categories, i.e., supervised and unsupervised. Taking advantage of the training sets, the supervised ones [9] are able to generate consistent results, however, the training process usually requires a large amount of manual labeling on the training models, which is tedious and time consuming. The unsupervised algorithms [1,2,3], on the other hand, are flexible and effective in segmenting shapes in a heterogenous shape database, however, these approaches either depend on accurate alignment of input shapes or cannot guarantee the consistency of the final consistency of the final co-segmentations across the whole set.

In this report, it is mainly about the research results of co-segmentation of 3D shapes via subspace clustering, and compares it with similar research. They are unsupervised co-segmentation for a set of shapes via descriptor-space spectral clustering by Sidi et al.[2], and unsupervised co-segmentation for 3D shapes using iterative multi-label optimization by Meng et al.[3].

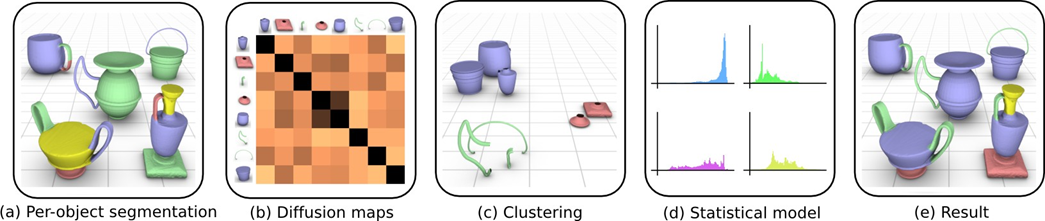


**Figure 1:***The co-segmentation result of the Candelabra set by our algorithm. Starting from the over-segmented patches of the shapes (left), our algorithm automatically obtains the consistent segmentations among these objects by grouping the patches using subspace clustering in multiple feature spaces. Corresponding parts are shown in the same colors (right).*

Ruizhen Hu et al. propose a novel unsupervised framework to consistently segment a set of 3D shapes from the same class. They consider the co-segmentation as a clustering problem by grouping the primitive patches of all shapes into corresponding parts (see Figure1). The core solution paradigm of their algorithm is a subspace clustering optimization [10] on the patches. Considering two parts of models perceived as corresponding may notnecessarily be similar in all features and may even significantly differ on some. They also propose a new subspace clustering formulation of optimization with a consistent multi-feature penalty to guarantee the consistency of co-segmentation results according to various features. By solving this optimization authors identify the most similar patch pairs consistent within multiple feature spaces and the prominent features contributing to measuring the similarity of each patch pair. Figure 2(upper left) shows the co-segmentation results with the consistent multi-feature penalty .



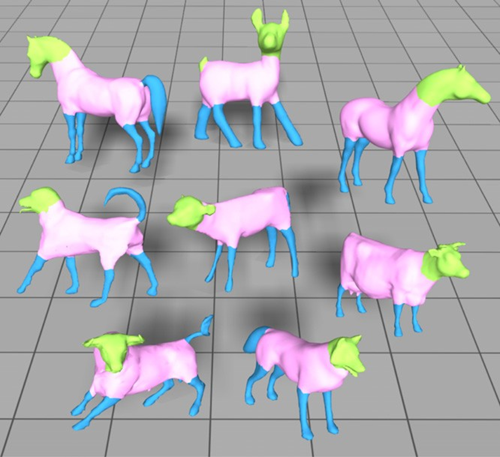
**Figure 2:***The legs of two table models are semantically in correspondence (upper left). They are quite similar in AGD features (upper right) while they differ a lot in SDF features (lower right). Hence, two parts of models perceived as corresponding may not necessarily be similar in all features. Upper left: result with the consistent multi-feature penalty; Lower left: result with the concatenated feature descriptor.*



**Figure 3:** *Overview of the steps in Sidi’s co-analysis: (a) An individual segmentation is computed for each shape. (b) The segments from all the shapes are embedded into a common space by using diffusion maps based on a similarity matrix, where darker colors in the matrix entries indicate higher similarities. (c) The segments are clustered in the embedded space. (d) A statistical model is created to describe each cluster.(e) The statistical model is used to label the shapes and obtain the final co-segmentation of the set*

The algorithm proposed by Sidi et al., like Ruizhen’s approach ,also based on an unsupervised method, and also treat the unsupervised co-segmentation of a set of shapes as a clustering problem. The clustering is performed in a space of shape descriptors rather than on the spatial coordinates of the shapes themselves. This allows the handling of corresponding on the statistical models. This step allows us to correct imperfections that appear in the initial co-segmentation, e.g., at segmentation boundaries. Figure 3 illustrates the steps of their algorithm. Showing that their unsupervised approach is able to co-segment families of shapes with significant geometric variations, achieving results that are competitive to supervised approaches.

Ruizhen points out that the correspondences between dissimilar parts could be built via linking through third-parties. However, Sidi’s approach needs initial segmentations for the shapes, which might result in un-satisfactory results when the per-shape segmentation cannot reflect the semantic parts well. On the other hand, their approach simultaneously generates the segmentations and their correspondences from the over-segmented patches, which is more flexible.



**Figure 4.** *Unsupervised co-segmentation results of our approach. Corresponding segments are shown with the same color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)*

Meng et al. propose a new framework for unsupervised co-segmentation of a set of 3D shapes. Our algorithm first segments all input shapes into the primitive patches, which are then clustered into corresponding parts to generate an initial estimation of co-segmentation. Next, the initial guess is automatically improved by an iterative optimization scheme, which generates a statistical model for the estimated clusters of all shapes and performs multi-labeling optimization for individual shapes respectively. Figure 4 shows their co-segmentation results on the class of four-leg animals. The experimental results demonstrate that their approach is able to achieve comparable performance to the supervised approach and produces better results than the unsupervised ones.

It is worth mentioning that both Ruizhen et al. and Meng et al. have chosen average geodesic distance (AGD)[10] and shape diameter function (SDF)[11] as feature descriptors in their algorithm, and Ruizhen has also added Gaussian curvature (GC) [12], shape contexts (SC) [13], and the conformal factor (CF) introduced in[14] for avoiding the result that only two shape descriptors to cluster the patches which might fail in cases of dissimilar objects.

[1] contributions are twofold.

* They propose a novel framework for shape co-segmentation based on subspace clustering. It simultaneously generates the segments and their correspondences from the over-segmented patches within multiple feature spaces. Various features can be introduced in our framework, which makes our algorithm flexible and feasible for co-segmenting different shapes.
* They propose a consistent multi-feature penalty in the subspace clustering optimization, which guarantees the consistency of the co-segmentation results according to various features and makes prominent features used to measure the similarity between each patch pair stand out actively.

# Related Work

A large variety of approaches have been proposed for segmenting single shape into meaningful parts. It has been shown [4] that no segmentation algorithm always performs well for all models because the geometry of an individual shape may lack sufficient cues to identify all parts that would be perceived as meaningful to a human observer.

**Co-segmentation of 3D shapes.** There have been various recent works on consistently segmenting a set of 3D shapes from the same class into semantic parts. Kraevoy et al. [15] perform an initial segmentation of each model into parts and then create a consistent segmentation by matching the parts and finding their correspondences. Huang et al. [16] present a linear programming based approach for jointly segmenting a heterogenous database of shapes, which significantly outperforms single-shape segmentation techniques. However, these two methods provide only mutually consistent segmentation but cannot guarantee the consistency of the final segmentations across the set. Golovinskiy and Funkhouser [17] consider the consistent segmentation as a graph clustering problem. By assuming that there is a global rigid alignment between matching shapes, their approach builds connection among the corresponding parts using ICP, and thus can handle limited model types only. To deal with non-homogeneous part scales, Xu et al. [18] classify the shapes based on their styles and then establish part correspondences in each style group. However, the graph generation process is computationally expensive. van Kaick et al. [19] incorporate prior knowledge learned from a set of pre-segmented and labeled shapes for performing part correspondences. Consistent segmentation may be established based on individual labeling of the shapes, however, a large number of manually segmented training shapes are needed in these learning based approaches, while Ruizhen’s approach is entirely unsupervised and does not require such training data.

**Subspace clustering.** Part of the research is inspired by the recent works of subspace clustering methods [20].Subspace clustering aims to cluster the high- dimensional datasets into multiplelow-dimensional linear subspaces simultaneously and has been widely used in computer vision, image processing, and data regression, etc.. Ruizhen treats the co-segmentation of a set of shapes as a subspace clustering problem within multiple feature spaces by introducing a consistent multi-feature penalty in the optimization.

**Spectral clustering.** Spectral clustering via diffusion maps is not new. The mesh segmentation work of de Goes et al. [21] specifically applies the diffusion distance, as Sidi et al. do in their work. All these solutions exploit special properties of the spectral embedding. However, the computed embeddings have always been a transformation from the spatial coordinates of individual input shapes. Sidi’s work computes spectral embeddings of shape descriptors. The descriptors of all the shapes in the set take part in the embedding, enabling him to perform a co-analysis.

**Shape descriptor.** Shape descriptors, characterizing the geometric features of input shapes, are widely used in digital geometry processing. Gatzke et al. [22] construct the curvature map signature for shape matching relying on geodesic distance. Lai et al. [23] propose a feature-sensitive metric by combing geodesic and isophotic distances for sampling, remeshing and multi-scale feature selection. Other metrics, such as shape distribution [24] and average geodesic field [25], have also been used to derive global shape properties in shape retrieval. Shape segmentation has to face the problem of defining a part or part boundary. As one of the best known rules, the minimal rule inducing part boundaries along negative curvature minima is typically realized via curvature measurements. Specific to shape segmentation , it is critical to find the proper metric which can capture the essence of the semantic components. Kalogerakis et al.’s [26] algorithm automatically selects the most informative features from a given set of shape descriptors to train the JointBoost classifier.

# Methodology

* 1. **Overview**

Ruizhen’s co-segmentation algorithm takes a set of meshes from an object category as input and produces a consistent segmentation of these meshes as output. First, they independently compute a set of primitive patches for each input mesh. Then, calculating a few feature vectors for each patch. Finally, they perform subspace clustering on all patches in multiple feature spaces and obtain co-segmentations of these meshes.

Sidi et al. make their co-analysis method takes as input a set of meshes from a given family and computes their co-segmentation and labeling. The co-analysis also provides a correspondence among the segments of any pair or group of shapes in the set, since the segments corresponding to the same part class will possess a common label. The label does not necessarily carry a semantic meaning, but serves more as a part index. For each family of shapes, the user also provides the maximumnumber of labels *L* that should be recovered from the set. This number loosely corresponds to the number of different kinds of semantic parts that constitute the shapes. Note that some types of parts can repeat or be omitted on the shapes, e.g., there can exist vases with multiple handles and vases without a base.

Meng et al. propose a new framework for unsupervised co-segmentation of a set of 3D shapes. Their algorithm first segments all input shapes into the primitive patches, which are then clustered into corresponding parts to generate an initial estimation of co-segmentation. Next, the initial guess is automatically improved by an iterative optimization scheme, which generates a statistical model for the estimated clusters of all shapes and performs multi-labeling optimization for individual shapes respectively.

Although both of them proposed algorithms are based on unsupervised methods for co-segmentation of 3D shapes, [2] has even greater limitations because it needs initial segmentations for the shapes, which might result in un-satisfactory results when the per-shape segmentation cannot reflect the semantic parts well. [1] and [3] have the most similar algorithms, but for [3], only two shape descriptors to cluster the patches which might fail in cases of dissimilar objects . On the other hand, [1] simultaneously generates the segmentations and their correspondences from the over-segmented patches, which is more flexible. And it choose more feature descriptors to be defined and computed on mesh triangles.

**Over-segmentation.** [1,3] perform an over-segmentation on each shape by partitioning it into primitive patches [16]. They employ the normalized cuts (NCuts) [27] to generate the primitive patches for each shape. And Meng et al.further align the patch boundaries with shape features by fuzzy cuts [28].

**Co-segmentation by subspace clustering.** In each feature space, the co-segmentation problem was treated as subspace clustering, thus algorithm can simultaneously generate the segmentations and their correspondences. All the patches in a single cluster represent a certain class of parts. To fuse multiple features, Ruizhen et al. propose a novel optimization formulation with a consistent multi-feature penalty. Our scheme facilitates both identification of most similar patches and identification of prominent features used for measuring the similarity between two similar patches. By considering all segments as primitives we can also perform subspace clustering on these segments to refine the co-segmentation results.

**Descriptor-space spectral clustering.** All the candidate segments produced by the per-object segmentation are embedded into a common space via diffusion maps. The embedding translates similarity between segments into spatial proximity, so that a clustering method based on Euclidean distance will be able to find meaningful clusters in this space. They obtain an initial co-segmentation by clustering the segments in the embedded space. All the segments in a single cluster potentially represent a certain class of parts, e.g., base, body, handle, or neck of a vase. This step is carried out with a hierarchical clustering scheme.

**Iterative multi-label optimization.** Given the initial guess of co-segmentation of shapes, Meng et al. build a statistical model to describe each cluster of parts, and employ the multi-label optimization scheme to produce the improved segmentation of input meshes. With the new estimate of co-segmentation, we can iteratively improve the segmentation of all shapes in the set. In place of a one- shot algorithm, our iterative scheme has the advantage of allowing automatic refinement of the co-segmentation. Specifically, the statistical model can be driven in terms of Gaussian mixture models (GMM) which are learned from the clusters via the shape.

* 1. **Algorithm**

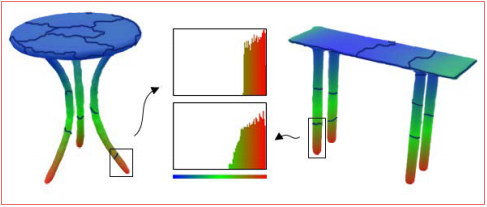
In the work related to the cooperative segmentation of a set of 3D shapes of the same type, [1] proposed that the algorithm of adding consistent punishment is not to connect features into high-latitude descriptors but to propose consistent multi-feature penalties in optimization, according to different The feature guarantees that the affinity matrix has the same sparsity. The modified algorithm can propose a consistent part structure throughout the model set. Therefore, in this report, the algorithm is described in detail.

**Notation.**Given n meshes M1, M2, ... , Mn from the same class, There are supposed to consistently segment each mesh into K meaningful parts. By over-segmentation,each mesh i is decomposed into pi patches, i = 1, 2, ... ,n. Thus there are  patches in total and have defined *H* feature vectors on each patch. For the *h*- th feature space (*h* = 1*,* 2*,* · · · *, H*), denote *xhi* as the feature vector of the *i*-th patch (*i* = 1*,* 2*,* · · · *, N*). Then get a feature matrix *Xh* = [*xh*1*, xh*2*, , xhN* ] for all patches in the *h*- th feature space (*h* = 1*,* 2*, , H*). Thus,the goal is to segment all patches into *K* clusters according to the feature matrices *X*1*, X*2*, , XH* , where each cluster defines the corresponding parts of the meshes.

· · ·

· · ·

· · ·



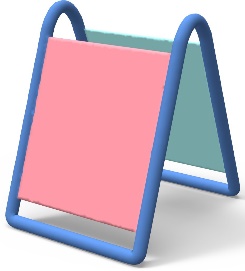
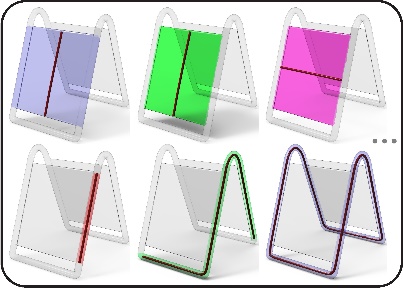
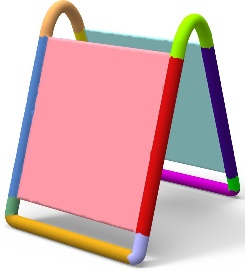
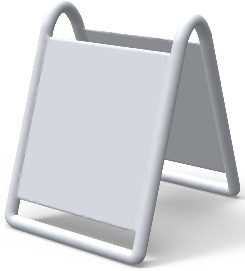
**Figure 4:** *Colormaps of AGD features of two tables with over-segmented patches. The AGD feature vectors of the two patches (marked in rectangles) from each table’s leg have similar distribution, as shown in histograms in the middle. We see that these two feature vectors lie in a common subspace generated by standard basis corresponding to the nonzero entries.*

We can see from Figure 3 that two corresponding patches have similar distributions (histograms), which means their feature vectors lie in the same subspace generated by standard basis corresponding to the nonzero entries. Therefore the task of segmenting the patches into parts can be considered as clustering their feature vectors {xh1, xh2,··· , xhN} in their respective subspaces. According to (4), the optimal coefficient matrix Wh ∈RN×N can be computed by solving: minWh F(Wh), subject to Wh ≥ 0 and diag(Wh) = 0, where

# Methodology

## Overview

Please first give the overview or pipeline of the algorithm. You are recommended to use figure or flow chart to illustrate the main idea of the paper. For example, Figure 1 shows the work flow of the algorithm proposed by [3].



* + 1. Input (b) Local GCs (c) Non-local GCs (d) Candidate GCs for ECP (e) Final Decom.

Figure 1: Our decomposition algorithm overview: given a shape (a), we first compute its local GC set (b), and then merge the local GCs into fewer overlapping non-local GCs (c) that form an over-complete cover for the shape (a). The final decomposition (e) is obtained by solving the exact cover problem from various candidate GC combinations (d) of non-local GCs (b).

## Algorithm

Please describe the paper’s algorithm in detail, including equations, pseudo-code, etc.

LATEX is a popular means by which to typeset complex mathematical formulae; it has been noted as one of the most sophisticated digital typographical systems in the world. But beginners often don’t know what to do. Some online tools can recognize hand-drawn and convert the formulas into LATEX code, for example, on *Web Equation*2, you can write your calculations or equations as shown in Figure 2, and get the beautiful formulas like equation 1.

*ei* = cos *θ* + *i* sin *θ* (1)

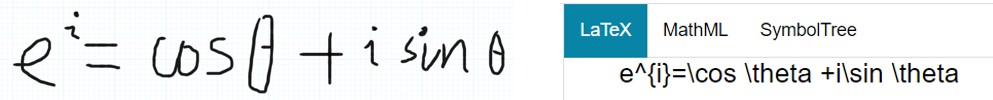


Figure 2: Web Equation

# Results

Please describe the experiment, including platform, test data, parameter, final results, etc.

2*webdemo.myscript.com/views/math.html*

Entering tables in LATEX documents can be burdensome because of the necessary formatting directives. There’re online generators which allow you to generate LATEX code that you can just copy & paste into your document’s source. For example, Table 1 shows the comparison on running time of different methods, which is generated from *tablesgenerator.com*3, as shown in Figure 3.

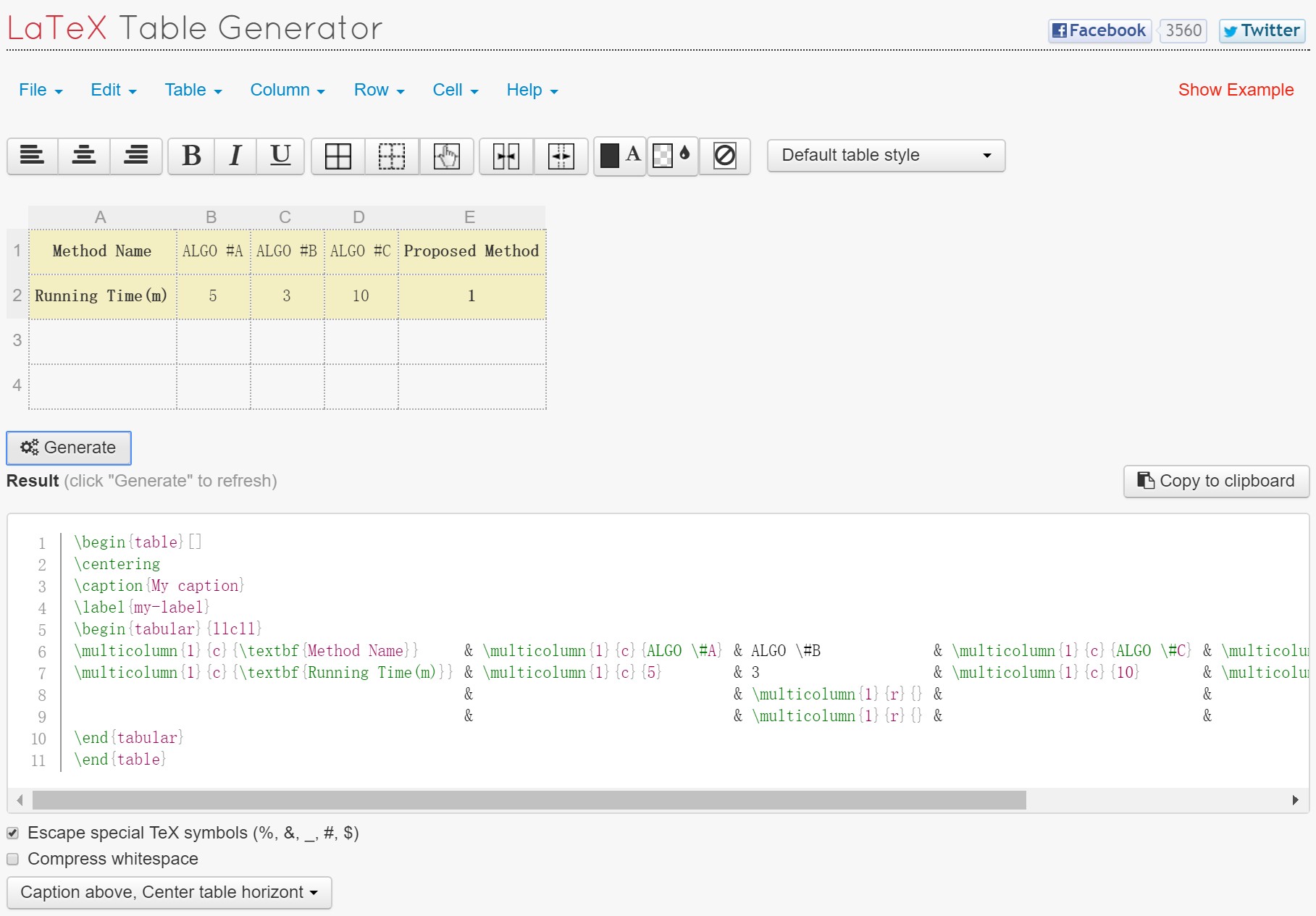


Figure 3: LATEX Table Generator

Table 1: The comparison on running time of different algorithms

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method Name** | ALGO #A | ALGO #B | ALGO #C | **Proposed Method** |
| **Running Time(m)** | 5 | 3 | 10 | **1** |

# Conclusion

Conclude the contribution of the paper, discuss the limitation and future research direction.

# Acknowledgement

You may write your acknowledgement here(optional).

3*[www.tablesgenerator.com](http://www.tablesgenerator.com)*

# References

1. Hu, R. , Fan, L. , & Liu, L. Co-segmentation of 3d shapes via subspace clustering. *Computer Graphics Forum,31*(5), (2012),1703-1713.
2. Sidi, O. , Van Kaick, O. , Kleiman, Y. , Zhang, H. , & Cohen-Or, D. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Trans. on Graph.* *(Proc. SIGGRAPH ASIA)* 30, 6 (2011), 126:1–9. [2](#_bookmark3), [3](#_bookmark6), [7](#_bookmark20), [8](#_bookmark23), [9](#_bookmark26), [10](#_bookmark30).
3. Meng M., Xia J., Luo J., He Y. Unsupervised co-segmentation for 3dshapes using iterative multi-label optimization. *Computer-Aided Design (Proc. ACM Symposium on Solid and Physical Modeling)* (2012). 3.
4. Attene, M. , G. Patané, & Al, E. Mesh segmentation -- A comparative study. *IEEE International Conference on Shape Modeling & Applications*. (2006), pp. 1–7.1,2.
5. Shamir, A. A survey on mesh segmentation techniques. *Computer Graphics Forum,* *27*(6), (2012),1539-1556.
6. Chen, X. , Golovinskiy, A. , & Funkhouser, T. A benchmark for 3d mesh segmentation. *ACM Transactions on Graphics,* *28*(3), (2009),1.
7. van Kaick O., Zhang H., Hamarneh G., Cohen-Or D. A survey on shape correspondence. In *Proc. of Eurographics State-of-the-art Report* (2010), pp. 1–24.2.
8. Huang Q., Koltun V., Guibas L. Joint-shape segmentation with linear programming. ACM Transactions on Graphics (Proc. SIGGRAPH ASIA) 30, 6 (2011), 125:1–11. 2,3.
9. GolovinskiyA., Funkhouser T. .Consistent segmentation of 3D models. *Computers and Graphics (Proc. Shape Modeling International)* 33, 3 (2009), 262–269. 2, 3.
10. Hilaga M., Shinagawa Y., Kohmura T., Kuniit. L. Topology matching for fully automatic similarity estima- tion of 3d shapes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), SIGGRAPH ’01, ACM, pp. 203–212. 2,3.
11. Shapirp L., Shalom S., Shamir A., COHEN-OR D., ZHANG H. Contextual part analogies in 3D objects. *International Journal of Computer Vision 89*, 2-3 (2010), 309–326. 2,3.
12. Gal R., Cohen-Or D. Salient geometric features for partial shape matching and similarity. *ACM Trans. Graph.* (2006), 130–150. 3.
13. Belongie S., Malik J., Puzicha J. Shape matching and object recognition using shape contexts. IEEE Trans. on Pattern Analysis and Machine Intelligence 24, 4 (2002), 509–522.3.
14. Ben-Chen M., Gotsman C. Characterizing shape using conformal factors. In *Proc. Eurographics Workshop on Shape Retrieval* (2008), pp. 1–8. 3.
15. Kraevoy V., Julius D., Sheffer A. Model composition from interchangeable components. In Proc. Pacific Graphics(2007), pp. 129–138. 3.
16. Huang Q., Koltun V., Guibas L. Joint-shape segmentation with linear programming. *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA) 30*, 6 (2011), 125:1–11. 2,3.
17. Golovinsky A., Funkhouser T. Randomized cuts for 3D mesh analysis. *ACM Transactions on Graphics (Proc. SIG GRAPH ASIA)* 27, 5 (2008), 145:1–10.3.
18. Xu K., Li H., Zhang H., Cohen-Or D., Xiong Y., Cheng Z. Style-content separation by anisotropic part scales. *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA) 29*, 5 (2010), 184:1–10.2, 3.
19. Van Kaick O., Tagllasacchi A., Sidi O., Zhang H., Cohen-Or D., Wolf L., , Hamarhen G. Prior knowldge for part correspondence. *Computer Graphics Forum (Proc. Eurographics)* 30, 2 (2011), 553–562.3*.*
20. Vidal R. A tutorial on subspace clustering. *IEEE Signal Processing Magazine 28*, 2 (2010), 52–68. 2,3,4.
21. De Goes, F., Goldenstein, S., and Velho, L. A hierarchical segmentation of articulated bodies. *Computer Graphics Forum (Proc. SGP) 27*, 5,( 2008), 1349–1356.
22. Gatzke T, Grimm C, Garland M, Zelinka S. Curvature maps for local shape comparison. *In shape modeling international*. (2005). p. 244–56.
23. Lai Y-K, Zhou Q-Y, Hu S-M, Wallner J, Pottmann H. Robust feature classification and editing. *IEEE Transactions on Visualization and Computer Graphics* (2007); 13(1):34–45.
24. Funkhouser T, Min P, Kazhdan M, Chen J, Halderman A, Dobkin D, Jacobs D. A search engine for 3D models. *ACM Transactions on Graphics* (2003);22(1): 83–105.
25. Gal R, Shamir A, Cohen-Or D. Pose-oblivious shape signature. *IEEE Transactions* *on Visualization and Computer Graphics* (2007);13:261–71.
26. Kalogerakis, E. , Hertzmann, A. , & Singh, K. . Learning 3D mesh segmentation and labeling. *ACM Trans. on Graphics (Proc. SIGGRAPH)* 29, 3 (2010), 102:1–10.2, 3, [8](#_bookmark23), [9](#_bookmark26), [10](#_bookmark30).
27. Golovinsky A., Funkhouser T. . Randomized cuts for 3D mesh analysis. *ACM Transactions on Graphics (Proc. SIG GRAPH ASIA)* 27, 5 (2008), 145:1–10.3.
28. Katz S, Tal A. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transactions on Graphics (Proc. SIGGRAPH)* (2003),22(3):954–61.