Automatic Vector Graphic Organization and Asset Extraction

Praveen Dhanuka
Adobe
dhanuka@adobe.com

Vineet Batra
Adobe
vbatra@adobe.com

Ankit Phogat
Adobe
phogat@adobe.com

Sumit Dhingra
Adobe
sudhingr@adobe.com

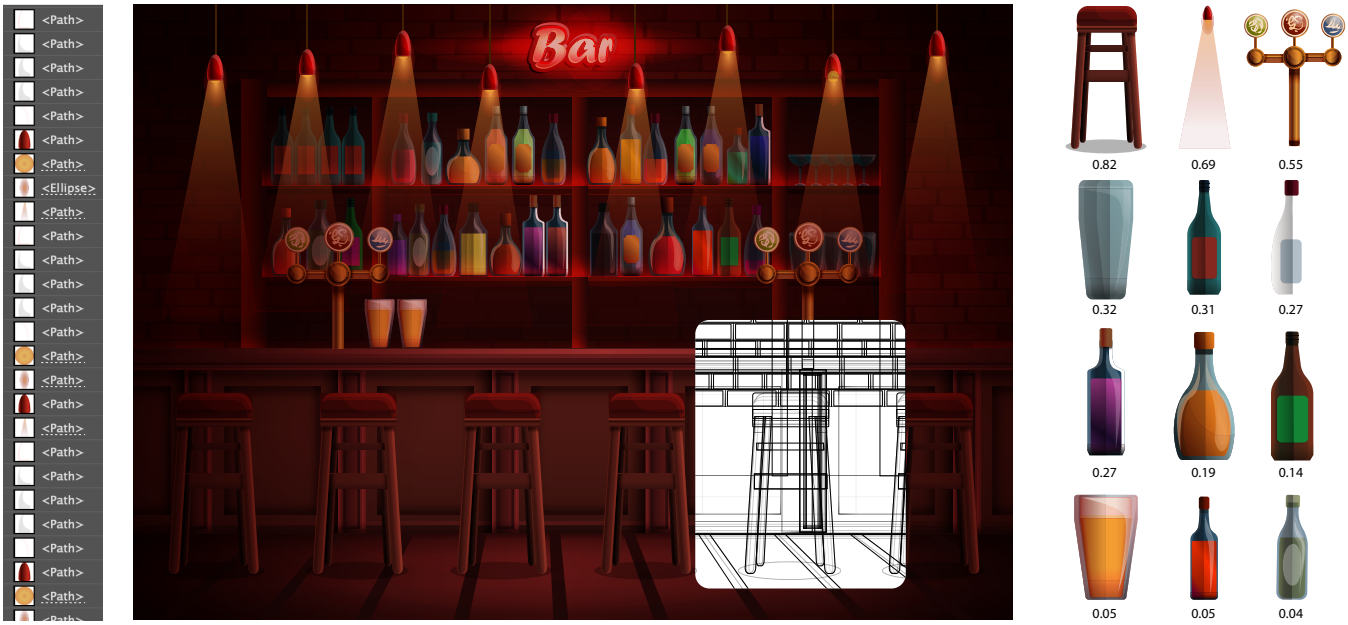

Figure 1: A real world vector illustration comprising of 3224 paths. *(Left)* Flat geometry in original illustration; *(Inset)* Underlying Bézier splines; *(Right)* Groups identified and their score.

ABSTRACT

We present a novel method for organizing Bézier bounded geometry based on affine similarity and visual saliency. For this computationally expensive (many-to-many) problem, we propose a highly parallel algorithm, leveraging programmable GPU pipeline, computing pairwise affine transformations to classify Bézier geometry into clusters, where paths in each cluster are affine transforms of each other. Using these clusters, we propose a method to organize paths into *meaningful* groups, even from complex, unstructured geometry, common in real-world vector art. We propose a function to quantify relative importance of these groups, using attributes such as complexity and frequency of occurrence, resulting in meaningful, reusable assets. Our method is both robust and performant, capable of processing thousands of paths within milliseconds.

CCS CONCEPTS

• **Computing methodologies → Computer graphics**.

SIGGRAPH '20 Posters, August 17, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-7973-1/20/08.
https://doi.org/10.1145/3388770.3407427

KEYWORDS

Assets Extraction, Vector Graphics, Clustering

ACM Reference Format:
Praveen Dhanuka, Vineet Batra, Ankit Phogat, and Sumit Dhingra. 2020. Automatic Vector Graphic Organization and Asset Extraction. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Posters (SIGGRAPH '20 Posters), August 17, 2020.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3388770.3407427

1 INTRODUCTION

Designers create rich vector graphic illustrations, which typically have highly complex geometry. When using creative applications, they often avoid non-creative and tedious steps, such as, organising their artwork into layer and groups or creating symbols for reuse. As a result, designers often end up with unstructured content that is hard to re-purpose. In fact, the geometrical structure is often incoherent making it difficult to organize and reuse content.

Recent advancements in deep learning have made it possible to identify and segment objects in raster images with reasonable accuracy [He et al. 2017]. But, this method does not translate well for vector graphics as it typically contain abstract, artistic shapes which do not find representation in natural images. Moreover, reverse mapping results from raster to vector domain does not work well (due to complex, overlapping geometry).Instead of translating to

raster domain, our method works directly with Bézier geometry and organizes vector graphics content into groups and identifying meaningful assets.

2 OUR APPROACH

Our algorithm has three stages; clustering, sequencing and ranking with input being an ordered sequence of cubic Bézier splines.

2.1 Clustering

In the first step, we cluster the input geometry based on affine variance. To achieve this, we customize Procrustes algorithm for 2D Bézier-bounded geometry and execute in parallel fashion to compute transformation matrix, among all $\binom{n}{2}$ pairs, using programmable GP-GPU pipeline. We then compute residue (l^2 norm), in parallel, using pairwise affine matrices and, in case residue is below a certain threshold ($1e^{-}3$), the paths are considered similar and added to the same cluster.

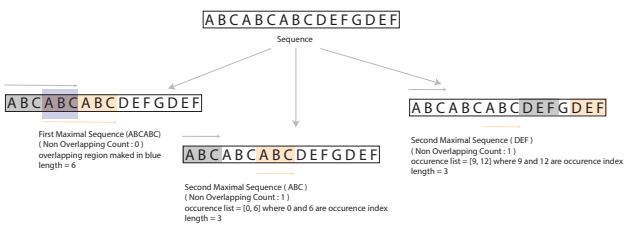

Figure 2: Maximally Repeating Patterns in sequence of mnemonics, corresponding to hierarchy of paths in input vector geometry. Paths belonging to same cluster are represented by single mnemonic.

2.2 Sequencing

We analyze the clusters of similar paths, and determine the sequence of paths repeating through these clusters in same pattern. Each path in cluster is assigned a mnemonic, and we extend biological sequencing algorithm to determine all *maximally repeating patterns*[Reinert et al. 2017] as illustrated in Figure 2.

Further, we eliminate sub-optimal patterns based on redundancy in sequence, in conjunction with the spatial characteristics of geometry corresponding to these mnemonic sequences. Specifically, we analyze the planar arrangement of geometry sequence and eliminate all sequences with disjoint faces or edges (Figure 3).

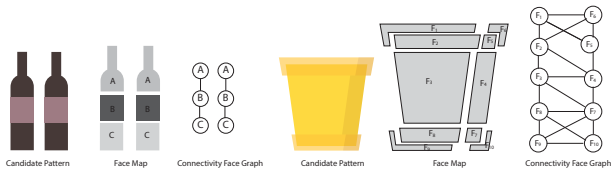

Figure 3: (*Left*) Group has disconnected faces and hence is eliminated. (*Right*) Group with single connected component

2.3 Ranking

We analyze each pattern(p) in the filtered set to compute its relative importance in the document. Following attributes are evaluated and a weighed combination is used to compute overall score.

2.3.1 Frequency(R_p). Frequency is the repetition count of a pattern in the given artwork.

$$R_p = (|p|) \tag{1}$$

2.3.2 Visual Saliency(V_p). Visual saliency denotes the visual importance of a pattern in the document and is computed by accumulating the area (*in pixels*) of all instances of the pattern in the document. Cumulative area can be approximated by 2 where s_{x_i}, s_{y_i} are scale factors of i_{th} repetition of pattern and A_b is the area of the *base* pattern.

$$V_p = \left(\sum_{i=1}^{C_p} \frac{s_{x_i} + s_{y_i}}{2} * A_b \right) \tag{2}$$

2.3.3 Geometric Complexity(G_p). This is total count of faces and edges in a pattern, after computing its planar arrangement [Asente et al. 2007]

$$G_p = (|F| + |E|) \tag{3}$$

2.3.4 Overall Score. The parameters R_p, G_p, V_p are normalized between [0, 1] and overall score of a pattern is computed using the weighted linear combination of normalized R_p, G_p, V_p as 4

$$\omega_p = \left(w_R * R_p + w_G * G_p + w_V * V_p \right) \tag{4}$$

3 RESULTS

The following table shows the results of our algorithm on a set of real-world vector graphics art. The extracted assets (shown in second column) are sorted on their overall score.

Table 1: *(Left)* Input vector graphic *(Right)* Groups identified using our method, ordered by score.

REFERENCES

Paul Asente, Mike Schuster, and Teri Pettit. 2007. Dynamic Planar Map Illustration. In *ACM SIGGRAPH 2007 Papers* (San Diego, California) *(SIGGRAPH '07)*. Association for Computing Machinery, New York, NY, USA, 30–es. https://doi.org/10.1145/1275808.1276415

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. 2017. Mask R-CNN. *CoRR* abs/1703.06870 (2017). arXiv:1703.06870 http://arxiv.org/abs/1703.06870

Knut Reinert, Temesgen Hailemariam Dadi, Marcel Ehrhardt, Hannes Hauswedell, Svenja Mehringer, Ren'e Rahn, Jongkyu Kim, Christopher Pockrandt, J"org Winkler, Enrico Siragusa, Gianvito Urgese, and David Weese. 2017. The SeqAn C++ template library for efficient sequence analysis: A resource for programmers. *Journal of Biotechnology* 261 (November 2017), 157–168.