Image-Space Collage and Packing with Differentiable Rendering

ZHENYU WANG, Shenzhen University, China
MIN LU*, Shenzhen University, China

Fig. 1. The 'SIGGRAPH' example is created using our method: (S) demonstrates the fundamental approach with good shape containment, non-overlap, and uniform distribution; (I) incorporates padding around geometric elements; (G-G) illustrates the smooth transition from axis-based initialization to final filling; (R) showcases collages with stripe blocks; (A) highlights packing within a shape with a downward force; and (P, H) display open packing arrangement with a downward force.

Collage and packing techniques are widely used to organize geometric shapes into cohesive visual representations, facilitating the representation of visual features holistically, as seen in image collages and word clouds. Traditional methods often rely on object-space optimization, requiring intricate geometric descriptors and energy functions to handle complex shapes. In this paper, we introduce a versatile image-space collage technique. Leveraging a differentiable renderer, our method effectively optimizes the object layout with image-space losses, bringing the benefit of fixed complexity and easy accommodation of various shapes. Applying a hierarchical resolution strategy in image space, our method efficiently optimizes the collage with fast convergence, large coarse steps first and then small precise steps. The diverse visual expressiveness of our approach is demonstrated through various examples. Experimental results show that our method achieves an order of magnitude speedup performance compared to state-of-the-art techniques.

CCS Concepts: • **Computing methodologies** → **Shape analysis**; **Image-based rendering**.

Additional Key Words and Phrases: Collage, Differetiable Rendering, Image Space

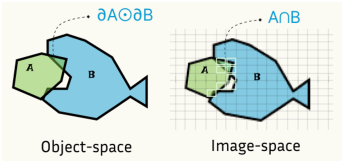
ACM Reference Format:
Zhenyu Wang and Min Lu. 2025. Image-Space Collage and Packing with Differentiable Rendering. In *Proceedings of (SIGGRAPH Conference Papers' 25)*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3721238.3730690

* Corresponding author: Min Lu (lumin.vis@gmail.com).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGGRAPH Conference Papers' 25, August 10-14, 2025, Vancouver, BC, Canada
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1540-2/2025/08
https://doi.org/10.1145/3721238.3730690

1 Introduction

Assembling and collaging geometric elements to encapsulate visual features provide a unified representation, which has been instrumental in creating intriguing visual designs and artworks, such as circular packing maps to show the thematic topic [4], word clouds for engaging overview of texts [33], or digital collages of photos [39]. Despite its popularity across various fields, the task of packing elements into given regions presents significant challenges. Numerous techniques have been proposed to address this task, with the majority of existing collage methodologies concentrating on *object-space* optimization [18, 28, 37, 46]. In object space, measuring the fit between geometric objects often involves designing geometric descriptors and energy functions specifically tailored to address the complexity of the objects' shapes.



Object-based techniques frame collages as a geometric constraint satisfaction problem, accompanied by certain limitations. First, geometric shapes usually need careful analysis to enable effective shape matching [18, 41]. For instance, reducing the overlap between shapes A and B necessitates the shape descriptors for their boundaries (∂A and ∂B). Additionally, geometric descriptors often lack generalizability. For instance, some works necessitate shapes with curvature and are unable to handle open shapes [18]. Some others are limited to fitting containers within convex boundaries [46]. Furthermore, the optimization process in object-based approaches can be computationally intensive, depending on the scale and complexity of the objects involved.

In this work, we advocate a paradigm shift in shape collage techniques by transitioning the geometric packing optimization from the object space to the image space. The core idea is to cast the geometric representation and their spatial relationships onto a grid of pixels, which are with fixed and object-independent complexity. Leveraging

the power of differentiable rendering [22], our method enables gradient backpropagation from image-space losses to geometric objects, effectively steering the collage optimization process from the discrete image space. Operating in image space facilitates a hierarchical resolution approach to dynamically manage the precision of these image-space losses. The collage process begins with low-resolution losses to facilitate large, bold adjustments and progressively increases resolution for finer refinements. This hierarchical approach significantly accelerates the computation, achieving an order-of-magnitude speedup compared to state-of-the-art methods.

The key strength of our technique is its ability to bypass complex object problem-solving by leveraging the inherent advantages of image-space optimization. This enables it to fit various geometric shapes into almost any desired target shapes. As shown in Figure 1, our method demonstrates its versatility by supporting a wide range of design configurations. These range from core space filling, as seen in 'S', to packing designs influenced by gravity effects in 'A', open-region packing exemplified by 'H', and complex shapes, such as the white stripe blocks in 'R'. Another advantage of our method, which employs gradual descent, is the smooth animation generated during the collage and packing process, as illustrated by the 'G's in Figure 1. In the evaluation, we compared our collage method against state-of-the-art baselines. Results show that our method significantly surpasses the baselines in visual quality. More importantly, our method achieves a remarkable improvement in computational efficiency and scalability, with gains on the order of magnitude.

2 Related Work

The collage and packing problem have been widely studied [48], including 3D object arrangement [10, 25, 55]. Below, we mainly review the research work related to 2D approaches.

Collage In 2D space, collage can be broadly categorized into two types: geometric graphics and images. Circular packing, exemplified by the work of Wang et al. [45], is a common paradigm, with new circles added to the outer periphery of existing ones. Several variations of circular packing have been developed, such as single-axis packing [24, 53], generative treemap [43], and hierarchical packing strategy [13]. Irregular shapes have also been considered, with methods like arclength descriptor matching [18] and autocomplete-based optimization [9]. Saputra et al. [37] represented objects as mass element meshes and used the repulsion forces between neighboring meshes to even out the negative space. Calligraphic packing, employed for letter composition, has been explored by Xu et al.[50] and enhanced for legibility by Zou et al.[56]. Those collage works deliberately describe primitives with geometric parameters, and then optimize over those parameters. Our approach avoids the need for complex geometric computation and the use of task-specific descriptors within the geometry space. Working in image space, our approach can be easily adapted for a variety of applications.

There is a another bunch of works achieving visually pleasing and balanced packing via a top-bottom manner, which divides the canvas into region cells via tessellation and then adjusts the placement of primitives within the cells. Kim and Pellacini [16] proposed an explicit packing energy function to optimize tiles for compact layouts. Hiller et al. [8] optimized the centroid placement of small objects such as dots and lines in the cells. Dalal et al. [3] proposed the Sum of Squared Distance metric for even distribution of the primitives with spatial extent. Further, Reinert et al. [34] facilitated real-time computation of the sum of squared distance using GPUs and allowed user customization by example. Unlike these methods with tiles and cell adjustments, our work optimizes primitives without global tessellation constraints. Primitives can overlap initially with overlaps, like the 'G' in Figure 1, and fit into open-shaped containers as shown in the 'PH' example of Figure 1.

Another related research topic is image collection, also called photo collage, which deliberately allows occlusions and blends. Many approaches have been proposed [36, 44]. For example, ShapeCollage [38] supports to interactively make a collage of photos with overlapping among photos. Rother et al. [36] allowed for soft intersection among photos. Goferman et al. [5] fused parts of photos into one whole image. Huang et al. [11] matched multiple cutouts from the Internet to compose a thematic figure. Liu et al. [23] extracted salient regions and proposed a correlation-preserved photo collage. Pan et al. [30] presented a content-based visual summarization technique for image collections. More recently, instead of matching existing photos, Lee et al. [20] generated collage artwork via reinforcement learning based on a given target image and materials, considering scores such as diversity, aesthetics, etc.

Text Filling Texts can be regarded as special geometric shapes. Considerable research has focused on arranging words to create text-based visual design and word art. Word clouds, popular for visually representing words in a compact layout, have been extensively studied [7, 35, 42]. Tools such as Wordle [27] help with the easy creation of word clouds. Cui et al.[2] proposed a dynamic force-directed model for word cloud layout, which preserves semantic context over time. Wu et al.[47] utilized seam carving to optimize word cloud layouts. Beyond traditional word clouds, researchers have explored filling words within specific shapes. Paulovich et al. [32] introduced a cutting-stock optimization method that optimizes the arrangement of words to maximize space utilization within shapes. ShapeWordle [46] took a different approach by utilizing the Archimedean spiral to accommodate irregular shapes, resulting in visually appealing word cloud compositions. MetroWordle [21] combined word clouds with maps, incorporating collision detection for geotags. Chi et al. [1] presented temporally morphable word cloud technology that allows word clouds to undergo smooth shape transformations over time. Xie et al. [49] proposed animating word cloud for emotional expression.

Some other works support interactive word cloud customization. For example, Koh et al.[17] introduced an interactive interface to facilitate user-driven word manipulation within word clouds. Jo et al. [15] introduced WordPlus, which expands the interaction of Wordle by incorporating pen and touch interactions. Additionally, Surazhsky et al.[40] proposed a method for text layout on 3D objects. Maharik et al.[26] used streamline-based techniques to arrange words artistically. Zhang et al. [52] introduced a word arrangement method that arranges theme-related words at the salient areas. Xu et al.[51] introduced a tone-based ASCII art generation method.

Unlike object arrangement guided by space-filling curves or collision detection, our framework utilizes image-based loss for flexible word fitting, accommodating loose compositions like force-attracted filling in a non-closed constrained boundary.

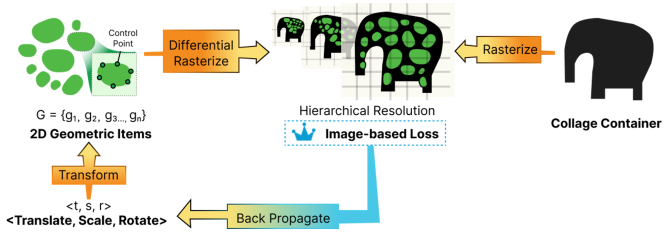


Fig. 2. Image-space collage and packing framework: starting with initialized 2D geometric items and their transformations, image-space losses are computed between the rasterized image and the target shape of the collage container across a hierarchy of image resolutions. These losses are then used to iteratively update the transformation parameters, refining the arrangement of the geometric items.

3 Preliminaries

Collage Problem. Given a set of 2D geometric items $G = \{g_1, g_2, \ldots, g_n\}$, the goal of collaging is to arrange them in a geometric container region C, where each shape s_i may undergo geometric transformations, including translate ($\mathbf{t}$), scale ($\mathbf{s}$) and rotate ($\mathbf{r}$). The optimization problem is formulated as:

$$\min_{\mathbf{t}_1,\mathbf{t}_2,\ldots,\mathbf{t}_n,\mathbf{r}_1,\mathbf{r}_2,\ldots,\mathbf{r}_n,\mathbf{s}_1,\mathbf{s}_2,\ldots,\mathbf{s}_n} \mathcal{L}(G, C, \mathbf{t}, \mathbf{r}, \mathbf{s}), \tag{1}$$

where $\mathcal{L}$ quantifies the arrangement quality within the container C. Non-overlapping and shape containment are two basic constraints:

$$\text{Shape Containment:} \quad G_i(\mathbf{t}_i, \mathbf{r}_i, \mathbf{s}_i) \subseteq C, \quad \forall i = 1, 2, \ldots, n$$
$$\text{Non-overlap:} \quad G_i(\mathbf{t}_i, \mathbf{r}_i, \mathbf{s}_i) \cap G_j(\mathbf{t}_j, \mathbf{r}_j, \mathbf{s}_j) = \emptyset, \quad \forall i \neq j$$

Vector Representation. We adopt a uniform vector representation for 2D geometric item of any shape. For each item, a closed area with N cubic Bézier curves $\{(p_x, p_y)\}_{i=1}^{3N}$ is initialized and fitted to the silhouette of the item via differentiable rendering. The parameter N controls the shape's granularity. In our examples, we set $N = 20$ to provide a balance of efficiency and geometric precision.

Differentiable Rendering. Rasterization can be considered as a mapping (or called scene function I) from the vector graphics to a 2D pixel grid, denoted as $I(x, y; \Theta)$, where (x, y) is the position of a pixel in the 2D grid, and Θ represents the vector graphic parameters (e.g., control points of a Bézier curve). Differentiable rendering is a class of techniques that makes the rasterization process differentiable. Differentiable rendering of vector graphics allows the backpropagation from the image domain to the vector graphics domain. Specifically, the scene function I is differentiated with respect to the parameters Θ. Several implementations exist, such as those using differentiable neural networks to approximate rasterization [29, 54]. In this work, we adopt the differentiable rendering approach by Li et al.[22], which leverages the observation that pixel colors become continuous after anti-aliasing.

4 Image-space Collage

Given the collage container C, the set of 2D geometric items G are iteratively optimized into a collage, as illustrated in Figure 2 for each epoch. First, each geometric item undergoes a geometric transformation applied to its control points P, resulting in:

$$\mathbf{P}'_i = \mathbf{r}_i \cdot (\mathbf{P}_i \odot \mathbf{s}_i) + \mathbf{t}_i, \quad \forall i, \tag{2}$$

where geometric items are continuously adjusted by parameters of $\mathbf{t}$ (translation), $\mathbf{s}$ (scaling), and $\mathbf{r}$ (rotation). Geometric items are then rasterized into an image $\hat{I}$ at resolution $w \times h$ by differentiable rendering. The container image is rasterized into a target image I_C, where the interior is black and the exterior is white. A series of image-based loss functions are calculated:

$$\mathcal{L}(\hat{I}(x, y; \mathbf{P}), I_C). \tag{3}$$

Following an update via gradient descent, the image loss is back-propagated to the parameters within the geometric transformation, updated as:

$$\mathbf{t}, \mathbf{s}, \mathbf{r} := \mathbf{t} - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{t}}, \ \mathbf{s} - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{s}}, \ \mathbf{r} - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{r}}. \tag{4}$$

Over the optimization process, the image-space loss is calculated among raster images at multiple levels of resolutions, starting with a low resolution and gradually moving to higher resolutions. The transformation parameters are updated iteratively until the arrangement of items converges to an optimal solution.

4.1 Initialization

We propose a skeleton-based initialization method for distributing geometric items within the shape container. We use the Medial Axis Transform (MAT)[19] to extract the skeleton of the target shape and calculate the medial width (distance to the nearest boundary point). As illustrated in Figure3, visual elements are distributed along the medial axis, with larger elements positioned at points with greater medial width. This approach ensures an even distribution of elements within the shape, making it especially effective for tubular shapes. Note that our method is robust to initialization. In Section 5, we show it can effectively handle poor initialization conditions as discussed in [28].

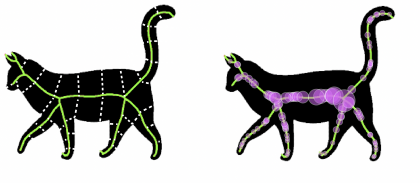
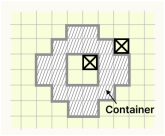


Fig. 3. MAT-based position initialization: with the detected medial axes and their nearest associated widths to the boundary (left), visual elements are initialized in the way that larger ones are placed on axes with larger medial widths (right).

4.2 Image-based Loss

We designed the image-space function to ensure two essential requirements, i.e., shape containment and non-overlapping.

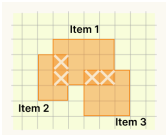
Shape Containment. We propose a spatial penalty mask to enhance the basic image mean square error loss (MSE loss) by encouraging elements to full fill the target shape. Geometric items are rendered into black and white image $\hat{I}_{b\&w}$. The



penalty mask $W \in \mathbb{R}^{w \times h}$ assigns a small penalty ($w_{ij} = 1$) for pixel difference within the target container region and a large penalty ($w_{ij} = 100$) for difference outside. The Weighted Mean Square Error (WMSE) between differentiable rasterized image $\hat{I}$ and the target image I is then calculated as:

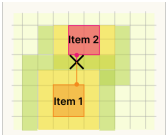
$$\mathcal{L}_{\text{containment}} = \frac{1}{w \cdot h} W \odot \left\| \hat{I}_{b\&c} - I_C \right\|^2. \tag{5}$$

Non-overlapping. In image space, detecting overlap among elements is straightforward and avoids geometric computations. Overlap is estimated by rendering all vector primitives with a fixed transparency τ, and and counting pixels whose transparency values deviate from τ, indicating overlapping regions, where $\mathbb{I}$ is an indicator function for the transparency condition, and p is a pixel of the image:



$$\mathcal{L}_{\text{overlap}} = \frac{1}{w \cdot h} \sum_{p} \mathbb{I}(T(p) > \tau). \tag{6}$$

Even Distribution. The two loss functions discussed above constrain visual elements within the target shape and prevent overlap, but uneven distribution may still occur, negatively impacting overall visual quality. To address this, we propose a uniform loss $\mathcal{L}_{\text{uniform}}$. This is achieved through differentiable image dilation (d), using a series of convolution kernels with increasing bandwidths (starting at five pixels, incrementing by six pixels per step) to approximate a distance field. As defined in Equation 7, $\mathcal{L}_{\text{uniform}}$ is computed as the weighted sum of pixels (p) in non-occupied regions within the collage container. The weights (w) are assigned based on kernel bandwidths, increasing for larger kernels. Larger dilations highlight broader gaps and are assigned higher weights, while smaller dilations receive lower weights. This approach emphasizes larger spaces, prioritizing their reduction to achieve a more uniform distribution. The equation is as follows:



$$\mathcal{L}_{\text{uniform}} = \sum_{d} \sum_{p} w_d. \tag{7}$$

The overall loss function, shown in Equation 8, uses weights α, β, and γ set to 3e3, 8e4, and 5e-4 respectively, to determine the relative contributions of the factors in the optimization process:

$$\mathcal{L} = \alpha \mathcal{L}_{\text{containment}} + \beta \mathcal{L}_{\text{overlap}} + \gamma \mathcal{L}_{\text{uniform}}. \tag{8}$$

4.3 Hierarchical Image Resolution

The resolution of the image $\hat{I} \in R^{w \times h}$ plays a crucial role in balancing loss precision and computational efficiency, as is also observed in general image analysis tasks [6]. Figure 4 illustrates this trade-off. Low-resolution images enable faster loss computation but provide lower precision in detecting overlap and containment, resulting in reduced collage quality. Conversely, high-resolution images enhance precision and overall collage quality but come with significantly higher computation costs. To balance precision and efficiency, we adopt a hierarchical strategy: starting with a low resolution (50×50) to expedite initial computations and progressively increasing to a high resolution (600 × 600) for refinement. Further details are discussed in Section 6.1.

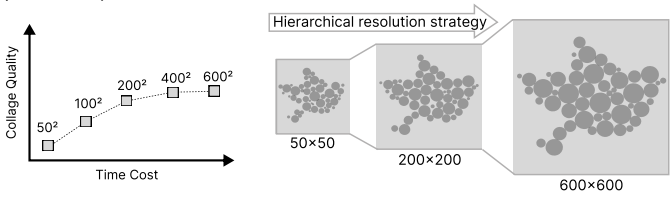


Fig. 4. Trade-off between collage quality and computation time for different image resolutions. Note that the three collages on the right have been resized for better visualization of quality differences and do not reflect their original resolution.

5 Results

Building on the core image-space collage method introduced earlier, Figure 13 presents examples of visual collage designs, spanning from intricate vector icons to hand-drawn sketches. Figure 14 demonstrates how the collage technique integrates seamlessly with images. Below, we explore how this technique can be extended to support a diverse range of use cases.

Force Attraction Our method can be seamlessly integrated with force-directed techniques. For example, by defining an attracting or repelling force source, the distance between visual elements and the force source can be computed as a loss function to influence the movement of the elements. This approach enables controlled attraction or repulsion of elements based on the specified force field. As shown in Figure 5 (left), a packing layout such as a circular or horizontal layout can be achieved using a central force point or a linear downward force. Additionally in Figure 5 (right), elements can be attracted into the mask by the force attraction with the collage mask.

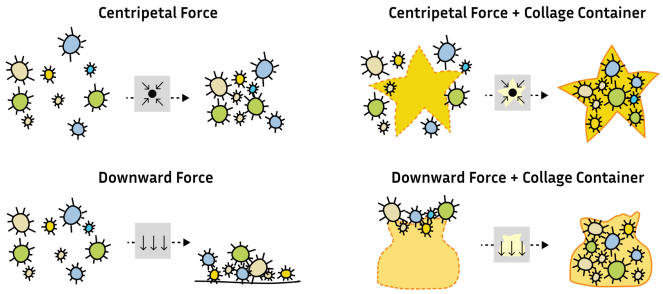


Fig. 5. Packing examples with attracting forces: (left) our technique integrates a centripetal or downward force to pack elements efficiently within an open area; (right) using a collage container, elements are first attracted and confined within specific shapes.

Animation Effects The gradual optimization process of our collage technique produces a side effect: captivating animation effects, distinguishing it from search-and-match algorithms [18]. As shown in Figure 6(top), the star glyphs are initialized on the top line and fall by a downward attracting force, creating an animation effect of 'falling down'. In Figure 6(bottom), by the MAT-based initialization, the visual elements move outwards to fit in the shape of the US boundary, creating an animation effect of 'expanding'.

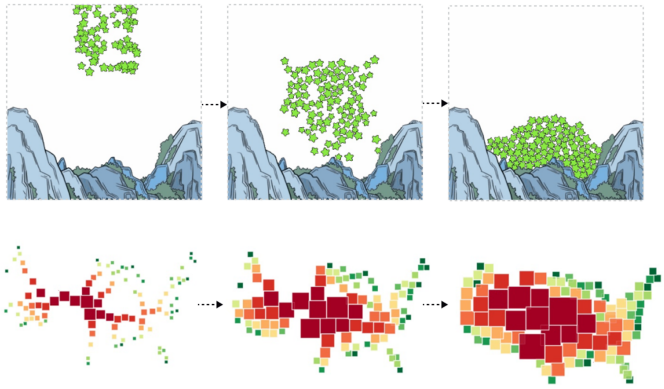


Fig. 6. Gradual optimization creates animation effects: (top) an expanding animation effect, (bottom) a falling down animation effect.

Fig. 7. Word clouds that pack words into animal shapes: from left to right, a vertically-aligned collage in the shape of a seahorse, a horizontally-aligned collage in the shape of a cat, and a loosely horizontally-aligned collage in the shape of a bird.

Graphic Text Blending Our method seamlessly integrates text and graphics, enabling cohesive and visually appealing compositions. As illustrated in Figure 7, our technique is used to generate word clouds (also known as wordles), where words of varying font sizes are arranged to form specific shapes, creating a balanced and engaging design. In Figure 14, we show examples of blending texts within image regions with low saliency. This ensures that less prominent areas are utilized effectively, enhancing the overall layout while maintaining the visual emphasis on key elements.

Data Visualization. Our method supports unit visualization, where each visual element represents a data item [31]. In Figure 8 (left), the Country Coffee Production dataset[1] is visualized, with each coffee-producing country represented by a coffee bean. The size of the bean encodes the country's coffee production, and the uniform scaling parameter **s** ensures accurate area-based encoding without loss of fidelity. Figure 8 (right) illustrates a bar-like sedimentary visualization of the Nobel Prize Winners in the US from 1902[2]. Each winner is represented by a hand-drawn circle, with colors indicating prize categories, creating a clear and engaging representation of the dataset. In this example, a downward force is integrated to create visual effect of sedimentation.

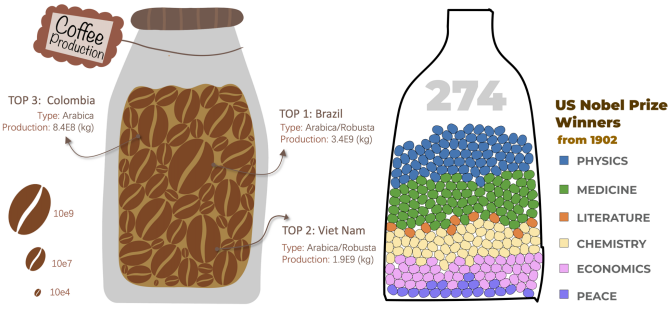


Fig. 8. Unit data visualization examples: (left) coffee production infographic, in which each coffee bean is a county that produces coffee, and its size encodes the coffee production. (right) Nobel prize winners in the US, each Nobel winner is represented as a circle, whose color indicates its category.

6 Evaluation

In this section, we first report the results of the ablation study and then elaborate on the comparison between our method and state-of-the-art methods.

Metrics of Collage Quality. We used three metrics to quantitatively measure the quality of the generated collages. The first metric is adopted from exiting work [46], and additional two metrics are added to quantify the overlaps among objects and the target shape: (1) *Layout Coverage (LC):* it is quantified as the proportion between the number of pixels in the object area (i.e., words in the clouds) and the number of pixels in the non-object area inside the target shape, the bigger the better; (2) *Object Overlap (OO):* it is quantified as the ratio of pixels in the overlap areas between objects to the total number of pixels in the target shape; (3) *Exceeding Area (EA):* it is quantified as the ratio of pixels that exceed the target shape to the total number of pixels in the target shape.

6.1 Ablation Study

Ablation on Uniform Loss. We investigated the impact of uniform loss on collage quality, and quantified the *Layout non-Uniformity (L-nU)* as the averaged distance square of non-object pixels to their nearest objects. As shown in Figure 9, *without the uniform loss*, the elements are less evenly distributed. For example, in the highlighted regions, collages without uniform loss leave noticeable gaps in other areas, which disrupt the overall balance and aesthetic consistency of the design.

Ablation on Image Resolution Strategies. We conducted an ablation study to evaluate the impact of different image resolution strategies on performance. The study examined eight constant resolution approaches, ranging from 50 to 1200 (shown in Table 1), and two hierarchical resolution strategies *100 + 600* and *50 + 200 + 600*.

The evaluation was conducted under four conditions, packing 100 elements into four different collage shapes to assess the effectiveness

[1] https://www.kaggle.com/datasets/michals22/coffee-dataset
[2] https://www.kaggle.com/datasets/joebeachcapital/nobel-prize

Fig. 9. Comparison of collages with and without the uniform loss: with uniform loss, the elements (e.g., texts on the right example) exhibit a more evenly distributed arrangement.

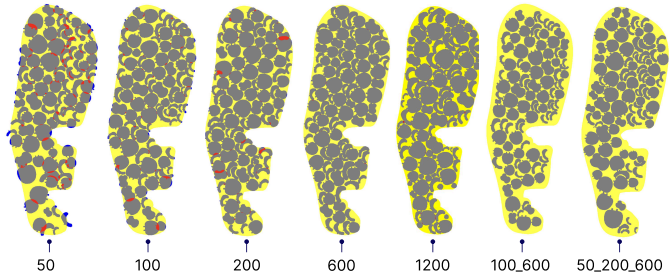


Fig. 10. Result samples generated using different resolution strategies: the left five use constant resolutions from low to high, while the right two employ hierarchical resolution strategies.

of resolution strategies across varying shape complexities. All examples are generated over 200 optimization epochs. For the hierarchical resolution strategy, the epochs were evenly distributed across each resolution level. Figure 10 shows collage samples generated using different image resolutions. In the figure, the target shapes are shown in yellow, visual objects in gray, overlaps between visual objects (OO) in red, and areas exceeding the target shapes (EA) in blue.

The averaged results are summarized in Table 1, revealing clear trade-offs between resolution strategy and time cost. As can be seen, collage quality improves as the resolution increases. The constant high-resolution 600×600 achieved high metric scores but incurred significant computational overhead. In contrast, the hierarchical strategies, especially 50 + 200 + 600, demonstrated more balanced performance. They delivered competitive metric results while maintaining lower time costs, highlighting their efficiency in managing resolution adaptively.

6.2 Comparison Experiment

We compared our method with four existing methods: PAD [18], Minkowski Penalty [28], ShapeWordle [46], and ShapeCollage [38]. ShapeWordle is designed specifically for text, while ShapeCollage is tailored for rectangular images. Due to the limitations of these methods in handling general geometric shapes, we performed one-on-one comparisons between our approach and each baseline in specific scenarios.

Qualitative Comparison. We compared our method with PAD and Minkowski Penalty, both designed for generating shape collages. As shown in Figure 11, our method delivers visually comparable results to both methods, while being significantly faster. Specifically, our method generates the example in six minutes, compared to over 700 minutes for PAD. Against Minkowski Penalty, our method demonstrates similar time efficiency for the tested examples, around 100 elements. As the number of elements increases, the optimization time for Minkowski Penalty would grow longer due to its $O(n^2)$ time complexity for arranging constraint pairs.

Table 1. Comparison of collage quality and time cost over resolution strategies: time is the cost of 200 optimizing epochs.

Resolution	Coverage (LC)	Overlap (OO)	Exceed (EA)	Time (s)
50x50	69.67%	2.72%	1.11%	12.90
100x100	65.60%	0.68%	0.33%	13.41
200x200	73.43%	0.42%	0.07%	15.44
400x400	67.55%	0.08%	0.01%	22.36
600x600	70.77%	0.11%	0.00%	32.54
800x800	72.23%	0.12%	0.00%	52.42
1000x1000	71.16%	0.05%	0.00%	76.04
1200x1200	69.39%	0.01%	0.00%	99.76
100+600	51.47%	0.02%	0.00%	18.51
50+200+600	60.73%	0.01%↓	0.00%	20.74 ↓

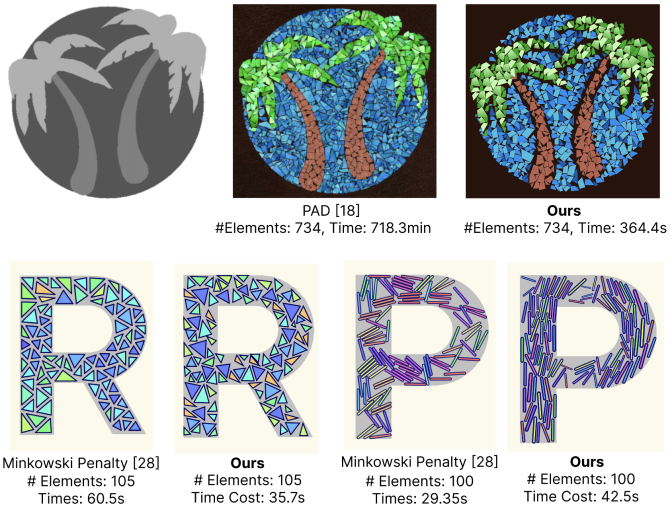


Fig. 11. Examples and their time cost generated by our method and others: (top) with PAD [18], and (bottom) with Minkowski Penalty [28].

Quantitative Comparison. We performed a quantitative comparison between our method and PAD, ShapeWordle and ShapeCollage, based on the three collage quality metrics. We tested ShapeWordle and our method on two target shapes ('flower' and 'leaf' in Figure 12) from the ShapeWordle website[3]. PAD and our method were tested using two examples from the PAD work [18] ('Australia' and 'Letter P'). For comparison with ShapeImage, two general target shapes were chosen, 'Moon' and 'Fish'. Table 2 summarizes the three metrics of different methods across the six examples.

[3]https://www.shapewordle.com/

Table 2. Comparison between our method and three baselines on the six examples in Figure 12.

Metrics \ Methods	Fig. 12 Flower		Fig. 12 Leaf		Fig. 12 Letter P		Fig. 12 Australia		Fig. 12 Moon		Fig. 12 Fish	
	ShapeW.	Ours	ShapeW.	Ours	PAD	Ours	PAD	Ours	ShapeC.	Ours	ShapeC.	Ours
Layout Coverage (LC)	0.25	**0.28** ↑	0.18	**0.29** ↑	0.94	0.80	0.91	0.75	0.67	**0.86** ↑	0.67	**0.87** ↑
Object Overlap (OO) $\times 10^{-3}$	0	0	0	0.02	33.83	**0.14** ↓	55.30	**0.30** ↓	41.44	**0.09** ↓	40.04	**0.08**↓
Exceeding Area (EA)$\times 10^{-3}$	0	0	0	0	7.17	**0** ↓	21.84	**0** ↓	18.31	0	7.86	**0** ↓

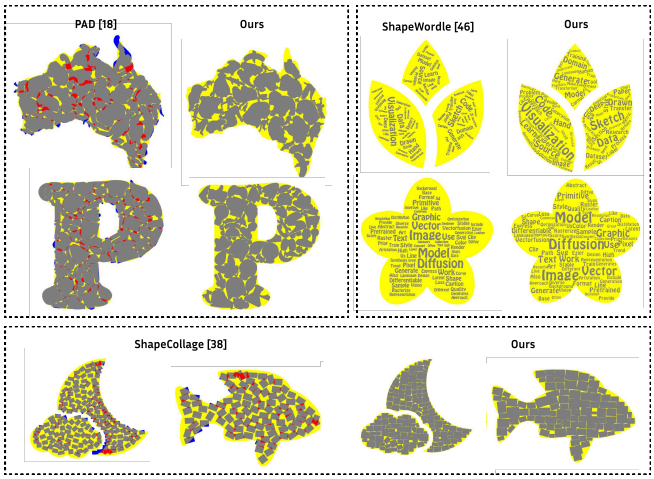


Fig. 12. Comparison between our method and three existing methods: yellow areas are the target shape, red areas are where visual objects overlap, and blue areas are where visual objects exceed the target shape.

Figure 12 uses the same visual encoding as Figure 10. As can be seen, compared to ShapeWordle, our method consistently demonstrates superior performance in Layout Coverage (LC). In both the 'leaf' and 'flower' examples, our method has much less space left, and the distribution is more even. Our method outperformed ShapeImage in all three metrics. As can be seen in the 'Moon' and 'Fish' examples, our method achieves larger coverage, but with much more even distribution, less overlap among objects, and less exceeding from the target shapes. As shown in the 'Australia' and 'Letter P' examples, PAD gets a more compact layout than ours, with less space left, which results in better scores in Layout Coverage. However, PAD causes more severe overlapping (i.e., the red and blue areas in Figure 12) than ours.

7 Conclusion and Future Work

In this work, we have introduced a neat approach to creating collage and packing visualizations by leveraging vector graphics manipulation through an optimization process aimed at minimizing loss in image space. Through the diverse examples presented in Section 5, we have demonstrated the versatility of our method in generating visually compelling collages. Compared to object-based methods such as PAD [18] and Minkowski Penalty [28], our method offers the advantages of being free from object-specific representations and achieving greater computational scalability. Our image-space approach empowers users to explore their creativity, experiment with novel visual elements, and incorporate imaginative concepts, thereby expanding the possibilities for expressive and engaging visual design.

Link with Image Generation Models In light of the advancements made, there are several promising directions for future research and development. One potential avenue is the exploration of interactive interfaces for target image-space editing in visualization creation. By providing users with intuitive editing and controls in the target image, they can directly manipulate and refine the visual elements in return, allowing for a more interactive and iterative design process. Designing an interactive system for collage authoring would be an interesting work in the future. A more promising avenue is to import text-driven editing for collage design based on a text-to-image foundation model [12][14].

Element Initialization. In this work, we experimented with one primitive initialization method, MAT-based. It is important to note that different primitive initialization methods can be suitable for different conditions, depending on the specific requirements and constraints of the application. For example, the MAT-based initialization proves effective for shapes with varying widths, such as tubes and necks. As seen in the 'tail of seahorse' of Figure 7, our experiments validate the promising results achieved through the MAT-based initialization technique. However, the MAT-based method is not optimal for target shapes with round bellies. Potential future work is to study adaptive primitive initialization techniques that automatically suggest initial visual primitives based on the geometric features of the target shape. This would enhance the efficiency and accuracy of the initialization process, leading to better adaptation of our method to diverse geometric configurations.

The Curse of Local Minima. Like any other iterative optimization algorithms with loss functions, our method can get stuck in some local minima, when visual primitives are not well-fitted in the target shape. When some small primitives are fully contained in some big elements, they are shadow-trapped. This obstruction leads to a state of stagnation, where the primitive remains stationary and unable to move. Some techniques can be used to alleviate the curse of local minima. For example, a sheepherder algorithm can be integrated into the collage optimization procedure, which can monitor and report problems in a global scope, such as detecting the coverage of visual elements, etc.

Hybrid Object- and Image-space In Figure 12, we demonstrate that our method outperforms other object-based approaches in collage generation, especially in terms of compactness and non-overlap. However, object-space methods have distinct advantages. For instance, methods like Minkowski Penalty [28] provide finer control over object properties, such as preserving balance and harmony among selected objects. A promising future direction would be to incorporate object-space loss into our framework to refine spatial relationships further and enhance layout quality.

Acknowledgments

We are deeply grateful to Prof. Daniel Cohen-Or and Prof. Dani Lischinski for their encouragement and insightful feedback throughout this work. We also thank the anonymous reviewers for their constructive suggestions. This work is supported in parts by fundings from Shenzhen Science and Technology Program (20231122121504001), National Natural Science Foundation of China (NSFC) Program (62472288), and Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), MNR Key Laboratory for Geo-Environmental Monitoring of Great Bay Area, and Guangdong Key Laboratory of Urban Informatics.

References

[1] Ming-Te Chi, Shih-Syun Lin, Shiang-Yi Chen, Chao-Hung Lin, and Tong-Yee Lee. 2015. Morphable word clouds for time-varying text data visualization. *IEEE transactions on visualization and computer graphics* 21, 12 (2015), 1415–1426.

[2] Weiwei Cui, Yingcai Wu, Shixia Liu, Furu Wei, Michelle X Zhou, and Huamin Qu. 2010. Context preserving dynamic word cloud visualization. In *2010 IEEE Pacific Visualization Symposium (PacificVis)*. IEEE, 121–128.

[3] Ketan Dalal, Allison W. Klein, Yunjun Liu, and Kaleigh Smith. 2006. A spectral approach to NPR packing. In *Proceedings of the 4th International Symposium on Non-Photorealistic Animation and Rendering* (Annecy, France) *(NPAR '06)*. Association for Computing Machinery, New York, NY, USA, 71–78. https://doi.org/10.1145/1124728.1124741

[4] Daniel Dorling. 2011. *Area Cartograms: Their Use and Creation*. Vol. 59. 252 – 260.

[5] Stas Goferman, Ayellet Tal, and Lihi Zelnik-Manor. 2010. Puzzle-like collage. In *Computer graphics forum*, Vol. 29. Wiley Online Library, 459–468.

[6] Yuqi Gong, Xuehui Yu, Yao Ding, Xiaoke Peng, Jian Zhao, and Zhenjun Han. 2021. Effective Fusion Factor in FPN for Tiny Object Detection. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 1159–1167. https://doi.org/10.1109/WACV48630.2021.00120

[7] Marti A Hearst, Emily Pedersen, Lekha Patil, Elsie Lee, Paul Laskowski, and Steven Franconeri. 2019. An evaluation of semantically grouped word cloud designs. *IEEE transactions on visualization and computer graphics* 26, 9 (2019), 2748–2761.

[8] Stefan Hiller, Heino Hellwig, and Oliver Deussen. 2003. Beyond Stippling — Methods for Distributing Objects on the Plane. *Computer Graphics Forum* 22, 3 (2003), 515–522. https://doi.org/10.1111/1467-8659.00699 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.00699

[9] Chen-Yuan Hsu, Li-Yi Wei, Lihua You, and Jian Jun Zhang. 2020. Autocomplete element fields. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[10] Wenchao Hu, Zhonggui Chen, Hao Pan, Yizhou Yu, Eitan Grinspun, and Wenping Wang. 2016. Surface Mosaic Synthesis with Irregular Tiles. *IEEE Transactions on Visualization and Computer Graphics* 22, 3 (2016), 1302–1313. https://doi.org/10.1109/TVCG.2015.2498620

[11] Hua Huang, Lei Zhang, and Hong-Chao Zhang. 2011. Arcimboldo-like collage using internet images. In *Proceedings of the 2011 SIGGRAPH Asia Conference*. 1–8.

[12] Shir Iluz, Yael Vinker, Amir Hertz, Daniel Berio, Daniel Cohen-Or, and Ariel Shamir. 2023. Word-as-image for semantic typography. *ACM Transactions on Graphics (TOG)* 42, 4 (2023), 1–11.

[13] Takayuki Itoh, Yumi Yamaguchi, Yuko Ikehata, and Yasumasa Kajinaga. 2004. Hierarchical data visualization using a fast rectangle-packing algorithm. *IEEE Transactions on Visualization and Computer Graphics* 10, 3 (2004), 302–313.

[14] Ajay Jain, Amber Xie, and Pieter Abbeel. 2023. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1911–1920.

[15] Jaemin Jo, Bongshin Lee, and Jinwook Seo. 2015. WordlePlus: expanding wordle's use through natural interaction and animation. *IEEE computer graphics and applications* 35, 6 (2015), 20–28.

[16] Junhwan Kim, Fabio Pellacini, et al. 2002. Jigsaw image mosaics. *ACM Transactions on Graphics* 21, 3 (2002), 657–664.

[17] Kyle Koh, Bongshin Lee, Bohyoung Kim, and Jinwook Seo. 2010. Maniwordle: Providing flexible control over wordle. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1190–1197.

[18] Kin Chung Kwan, Lok Tsun Sinn, Chu Han, Tien-Tsin Wong, and Chi-Wing Fu. 2016. Pyramid of arclength descriptor for generating collage of shapes. *ACM Trans. Graph.* 35, 6 (2016), 229–1.

[19] Der-Tsai Lee. 1982. Medial axis transformation of a planar shape. *IEEE Transactions on pattern analysis and machine intelligence* 4 (1982), 363–369.

[20] Ganghun Lee, Minji Kim, Yunsu Lee, Minsu Lee, and Byoung-Tak Zhang. 2023. Neural collage transfer: Artistic reconstruction via material manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2394–2405.

[21] Chenlu Li, Xiaoju Dong, and Xiaoru Yuan. 2018. Metro-wordle: An interactive visualization for urban text distributions based on wordle. *Visual Informatics* 2, 1 (2018), 50–59.

[22] Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. 2020. Differentiable vector graphics rasterization for editing and learning. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–15.

[23] Lingjie Liu, Hongjie Zhang, Guangmei Jing, Yanwen Guo, Zhonggui Chen, and Wenping Wang. 2017. Correlation-preserving photo collage. *IEEE transactions on visualization and computer graphics* 24, 6 (2017), 1956–1968.

[24] Shixia Liu, Jialun Yin, Xiting Wang, Weiwei Cui, Kelei Cao, and Jian Pei. 2015. Online visual analytics of text streams. *IEEE transactions on visualization and computer graphics* 22, 11 (2015), 2451–2466.

[25] Y. Ma, Z. Chen, W. Hu, and W. Wang. 2018. Packing Irregular Objects in 3D Space via Hybrid Optimization. *Computer Graphics Forum* 37, 5 (2018), 49–59. https://doi.org/10.1111/cgf.13490 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13490

[26] Ron Maharik, Mikhail Bessmeltsev, Alla Sheffer, Ariel Shamir, and Nathan Carr. 2011. Digital micrography. *ACM Transactions on Graphics (TOG)* 30, 4 (2011), 1–12.

[27] Carmel McNaught and Paul Lam. 2010. Using Wordle as a supplementary research tool. *Qualitative Report* 15, 3 (2010), 630–643.

[28] Jiří Minarčík, Sam Estep, Wode Ni, and Keenan Crane. 2024. Minkowski penalties: Robust differentiable constraint enforcement for vector graphics. In *ACM SIGGRAPH 2024 Conference Papers*. 1–12.

[29] Reiichiro Nakano. 2019. Neural painters: A learned differentiable constraint for generating brushstroke paintings. *arXiv preprint arXiv:1904.08410* (2019).

[30] Xingjia Pan, Fan Tang, Weiming Dong, Chongyang Ma, Yiping Meng, Feiyue Huang, Tong-Yee Lee, and Changsheng Xu. 2019. Content-based visual summarization for image collections. *IEEE transactions on visualization and computer graphics* 27, 4 (2019), 2298–2312.

[31] Deokgun Park, Steven M Drucker, Roland Fernandez, and Niklas Elmqvist. 2017. Atom: A grammar for unit visualizations. *IEEE transactions on visualization and computer graphics* 24, 12 (2017), 3032–3043.

[32] Fernando V Paulovich, Franklina MB Toledo, Guilherme P Telles, Rosane Minghim, and Luis Gustavo Nonato. 2012. Semantic wordification of document collections. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 1145–1153.

[33] Andrew Ramsden and Andrew Bate. 2008. Using word clouds in teaching and learning. (2008).

[34] Bernhard Reinert, Tobias Ritschel, and Hans-Peter Seidel. 2013. Interactive by-example design of artistic packing layouts. *ACM Transactions on Graphics (TOG)* 32, 6 (2013), 1–7.

[35] Anna W Rivadeneira, Daniel M Gruen, Michael J Muller, and David R Millen. 2007. Getting our head in the clouds: toward evaluation studies of tagclouds. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 995–998.

[36] Carsten Rother, Lucas Bordeaux, Youssef Hamadi, and Andrew Blake. 2006. Autocollage. *ACM transactions on graphics (TOG)* 25, 3 (2006), 847–852.

[37] Reza Adhitya Saputra, Craig S Kaplan, and Paul Asente. 2019. Improved deformation-driven element packing with repulsionpak. *IEEE transactions on visualization and computer graphics* 27, 4 (2019), 2396–2408.

[38] ShapeCollage. [n. d.]. ShapeCollage. http://www.shapecollage.com/. Accessed: 2024-07-08.

[39] Yvonne Spielmann. 1999. Aesthetic features in digital imaging: collage and morph. *Wide Angle* 21, 1 (1999), 131–148.

[40] Tatiana Surazhsky and Gershon Elber. 2002. Artistic surface rendering using layout of text. In *Computer Graphics Forum*, Vol. 21. Wiley Online Library, 99–110.

[41] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. 2011. A survey on shape correspondence. In *Computer graphics forum*, Vol. 30. Wiley Online Library, 1681–1707.

[42] Fernanda B Viegas, Martin Wattenberg, and Jonathan Feinberg. 2009. Participatory visualization with wordle. *IEEE transactions on visualization and computer graphics* 15, 6 (2009), 1137–1144.

[43] Roel Vliegen, Jarke J Van Wijk, and Erik-Jan van der Linden. 2006. Visualizing business data with generalized treemaps. *IEEE Transactions on visualization and computer graphics* 12, 5 (2006), 789–796.

[44] Jingdong Wang, Long Quan, Jian Sun, Xiaoou Tang, and Heung-Yeung Shum. 2006. Picture collage. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 1. IEEE, 347–354.

[45] Weixin Wang, Hui Wang, Guozhong Dai, and Hongan Wang. 2006. Visualization of large hierarchical data by circle packing. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 517–520.

[46] Yunhai Wang, Xiaowei Chu, Kaiyi Zhang, Chen Bao, Xiaotong Li, Jian Zhang, Chi-Wing Fu, Christophe Hurter, Oliver Deussen, and Bongshin Lee. 2019. Shapewordle: tailoring wordles using shape-aware archimedean spirals. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2019), 991–1000.

[47] Yingcai Wu, Thomas Provan, Furu Wei, Shixia Liu, and Kwan-Liu Ma. 2011. Semantic-preserving word clouds by seam carving. In *Computer Graphics Forum*, Vol. 30. Wiley Online Library, 741–750.

[48] Gerhard Wäscher, Heike Haußner, and Holger Schumann. 2007. An improved typology of cutting and packing problems. *European Journal of Operational Research* 183, 3 (2007), 1109–1130. https://doi.org/10.1016/j.ejor.2005.12.047
[49] Liwenhan Xie, Xinhuan Shu, Jeon Cheol Su, Yun Wang, Siming Chen, and Huamin Qu. 2023. Creating emordle: Animating word cloud for emotion expression. *IEEE Transactions on Visualization and Computer Graphics* (2023).
[50] Jie Xu and Craig S Kaplan. 2007. Calligraphic packing. In *Proceedings of Graphics Interface 2007*. 43–50.
[51] Xuemiao Xu, Linling Zhang, and Tien-Tsin Wong. 2010. Structure-based ASCII art. In *ACM SIGGRAPH 2010 papers*. 1–10.
[52] Junsong Zhang, Zuyi Yang, Linchengyu Jin, Zhitang Lu, and Jinhui Yu. 2022. Creating Word Paintings Jointly Considering Semantics, Attention, and Aesthetics. *ACM Transactions on Applied Perceptions (TAP)* 19, 3 (2022), 1–21.
[53] Jian Zhao, Nan Cao, Zhen Wen, Yale Song, Yu-Ru Lin, and Christopher Collins. 2014. # FluxFlow: Visual analysis of anomalous information spreading on social media. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 1773–1782.
[54] Ningyuan Zheng, Yifan Jiang, and Dingjiang Huang. 2018. Strokenet: A neural painting environment. In *International Conference on Learning Representations*.
[55] Qiubing Zhuang, Zhonggui Chen, Keyu He, Juan Cao, and Wenping Wang. 2024. Dynamics simulation-based packing of irregular 3D objects. *Computers Graphics* 123 (2024), 103996. https://doi.org/10.1016/j.cag.2024.103996
[56] Changqing Zou, Junjie Cao, Warunika Ranaweera, Ibraheem Alhashim, Ping Tan, Alla Sheffer, and Hao Zhang. 2016. Legible compact calligrams. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.

Fig. 13. A gallery of examples: diverse visual elements (i.e., icons, sketched paths) can be effectively fitted within convex and concave target boundaries.

Fig. 14. A gallery of examples that texts and graphics are collaged and packed for visually appealing design.