DOI: 10.1111/cgf.70259
Pacific Graphics 2025
M. Christie, N. Pietroni, and Y.-S. Wang
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 44 (2025), Number 7

Using Saliency for Semantic Image Abstractions in Robotic Painting

Michael Stroh[1] Patrick Paetzold[1] Daniel Berio[2] Rebecca Kehlbeck[1] Frederic Fol Leymarie[2] Oliver Deussen[1] Noura Faraj[3]

[1]University of Konstanz, Germany
[2]Goldsmiths, University of London, United Kingdom
[3]LIRMM, CNRS, Université de Montpellier, France

(a) **(b)** **(c)** **(d)** **(e)** **(f)**

Figure 1: *Semantically guided image abstraction: (a) Original image; (b) combined saliency map generated by our method; (c) resulting enhanced abstraction with varying degrees of abstraction (facades, ground, windows on tower); (d) color quantized abstraction mapped to customized palette (15 colors); (e) painting by robot using oil colors on canvas; (f) e-David robot and painting*

Abstract

We present an adaptive, semantics-based abstraction approach that balances aesthetic quality and structural coherence within the practical constraints of robotic painting. We apply panoptic segmentation with color-based over-segmentation to partition images into meaningful regions aligned with semantic objects, while providing flexible abstraction levels. Automatic parameter selection for region merging is enabled by semantic saliency maps, derived from Out-of-Distribution segmentation techniques in combination with machine learning methods for feature detection. This preserves the boundaries of salient objects while simplifying less prominent regions. A graph-based community detection step further refines the abstraction by grouping regions according to local connectivity and semantic coherence. The runtime of our method outperforms optimization-based image vectorization methods, enabling the efficient generation of multiple abstraction levels that can serve as hierarchical layers for robotic painting. We demonstrate the quality of our method by showing abstraction results, robotic paintings with the e-David robot, and a comparison to other abstraction methods.

CCS Concepts

• ***Computing methodologies*** → ***Non-photorealistic rendering;*** *Image processing;* • ***Applied computing*** → *Fine arts;*

1. Introduction

Non-photorealistic rendering (NPR) [KCWI13,Her10] encompasses a broad range of techniques that transform images into abstract, artistic renditions, including image and video stylization, stroke-based rendering, and physical media simulation [GG01,RC13]. One particular aspect focuses on creating painterly renderings, which can even be realized as physical artworks inspired by the painting process of human artists. Robotic painting systems extend the original digital NPR techniques to the physical domain by realizing abstracted renderings as tangible artworks [LPD13]. By developing various painting styles, these systems provide insights into artistic processes, human-like conceptualization, and craftsmanship. Human painting processes are often poorly documented, and instead rely on artists' intuition [GD22].

Most image abstraction methods focus on creating digital renderings, prioritizing the aesthetic flexibility and appeal of the result without considering the limitations of realizing abstractions in the physical world. Similarly, digital, generative image models like

DALL·E [RPG*21], Stable Diffusion [HJA20], and Flux [Lab24] only generate pixel-based illustrations. However, compared to pixel-based approaches, robotic painting systems pose the additional challenge of requiring instructions, actions, and control mechanisms that perform brush strokes with physical materials, such as various paint types and colors, on different surfaces. These constraints include a limited number of available pigments, the precision and size of brushes, the precision of robot movement, drying times for paint, and the need to complete the painting within a reasonable timeframe.

Existing robotic painting approaches often rely on iterative stroke-based styles, such as ballpoint [TL12, TF13], ink-pen [LPD13], or layered brushstrokes [SMO23], where semi-transparent strokes are overlaid until the target color is reached. In contrast, we pursue structured, geometric abstractions inspired by Pop Art and Orphism, emphasizing color and composition over individual strokes. Our method organizes visual elements into coherent regions that provide high-level scene representations [Mou12, SGD23, GPD20], better reflecting the spatial relationships required for robotic execution. However, we need to ensure that abstract regions do not cross object boundaries, as this reduces semantic recognizability and can render important elements unidentifiable. To be realizable in a physical painting process, such abstractions must further respect specific constraints. Overly small or geometrically complex regions may be infeasible with available brush sizes, as thin strokes and sharp corners cannot be executed reliably and lead to unpredictable paint dynamics. Excessive overlaps create unplanned color interactions, while too many regions prolong execution, making the process impractical. Within these requirements, maximizing region size and minimizing their number shortens painting time and yields more controlled stroke placement. Importantly, large uniform regions in the abstraction need not appear flat in the final painting, as the robotic filling process naturally reintroduces texture and variation through the brush and paint dynamics, ensuring visual richness in the physical result.

Our approach builds upon the work of Stroh et al. [SGD23]. Their approach uses only panoptic segmentation and a shape tree data structure [FXDG17] to merge regions based on fixed color thresholds. However, in their work, extensive manual tuning is required to determine suitable abstraction levels, particularly in scenes with numerous objects, where thresholds must often be adjusted for each object. Furthermore, color-based merging tends to oversimplify large regions, collapsing key areas into single shapes while leaving isolated highlights. This can result in either overly detailed abstractions, with many small regions, or excessively simplified ones, where structural details critical to object recognition are lost.

We propose a novel image abstraction pipeline that transforms pixel-based input images into vectorized abstractions tailored to robotic systems. Similar to [SGD23], we use panoptic segmentation masks to introduce semantics into our abstraction process. As these masks are pretty coarse, they do not capture many image details, which might get lost in the abstraction. Therefore, to prevent this loss of detail, we propose as a key contribution of our approach the integration of additional semantic saliency masks and depth cues, capturing finer details like salient edges, enabling the incorporation of high-level scene understanding and semantic details into the abstraction process.

By combining panoptic segmentation with saliency predictions and additional feature extraction models, we estimate object-level importance to guide region merging. This allows the method to preserve semantically meaningful structures while simplifying less relevant areas in a controlled manner. Leveraging these features, our method addresses the practical challenges of robotic painting while minimizing the need for manual parameter tuning. Beyond robotic painting, the generated abstractions are also well-suited for image vectorization tasks. By partitioning the image into semantically coherent regions, our framework can convert pixel images into scalable vector graphics. We demonstrate how these abstractions enable robotic systems to produce artistic paintings on physical canvases, advancing the state of the art at the intersection of image abstraction and robotic art.

In summary, our contributions are:

- The integration of semantic saliency and region-based segmentation to balance detail and abstraction.
- A novel image abstraction pipeline transforming pixel-based images into vector representations optimized for robotic painting.
- An adaptive region merging process that enables automatic parameter selection while minimizing the need for manual tuning across diverse image types and abstraction levels.
- The implementation and demonstration of robotic systems like the e-David robot that reproduce the generated abstractions as artistic paintings on physical canvases.

2. Related Work

Segmentation and Abstraction A common strategy for image segmentation is a two-stage process: initial over-segmentation followed by region merging based on similarity. Felzenszwalb and Huttenlocher [FH04] introduced an efficient graph-based method based on internal variation, producing segments that align well with color boundaries. SLIC [ASS*10] groups pixels using color and spatial proximity, with extensions incorporating manual strokes [ZWQL16] or machine learning for improved edge detection [CLH*16]. While effective in grouping low-level features, these methods typically lack semantic awareness and produce overly fragmented or unstructured outputs unsuitable for robotic painting. To improve perceptual consistency, other approaches incorporate texture [HP10] or shape [XCGN16]. Hierarchical abstraction methods [FXDG17] and region merging in perceptual color spaces, such as CIELAB [SAM17], have been applied to stylization, typically with a fixed abstraction level for the entire image. We adopt the Felzenszwalb-Huttenlocher method for initial segmentation due to its unsupervised nature and strong alignment with perceptual boundaries, and enhance it with semantic-aware graph filtering for locally adaptive abstraction.

Learning-Based Segmentation and Vectorization Modern segmentation models, particularly panoptic methods [KHG*19], provide fine-grained object-aware outputs by combining semantic and instance segmentation. OneFormer [JLC*23] offers unified, high-quality panoptic predictions without prompt tuning, which we use to support object-level abstraction. To guide structural abstraction, we further incorporate monocular depth cues [YKH*24] and saliency maps derived from model uncertainty [WJ24]. For vectorization, differentiable methods like DiffVG [LLGRK20], LIVE [MZX*22],

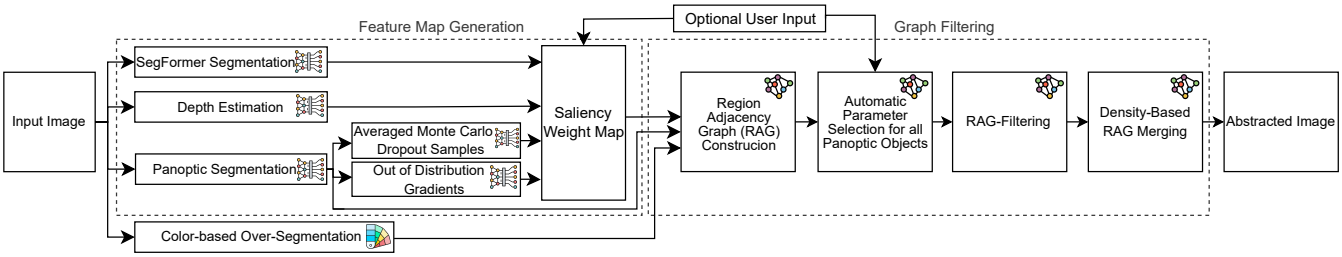


Figure 2: *Our proposed pipeline consists of three main parts: A feature map generation using machine learning and color based methods, a color difference based initial over-segmentation and a subsequent graph filtering step using graph based measures. The individual outputs of the semantic segmentation models and the combined saliency weight map are shown in Figure 3*

and others [ZZL24, HJA24] optimize closed curves using gradient descent. While powerful for digital stylization, they produce overlapping shapes and are computationally expensive – issues that hinder their use in robotic painting. Our method instead generates clean, non-overlapping, semantically-structured regions more amenable to physical reproduction.

Adaptive Abstraction and Parameter Selection Most existing segmentation-based abstraction methods rely on global parameters and treat the image uniformly, which limits their ability to preserve detail in important regions while simplifying others. Although some frameworks incorporate semantic labels [SGD23], they often require extensive manual tuning. Earlier work already emphasized that segmentation-based abstraction methods relied heavily on manual parameter tuning [PV08], which motivated research into reducing this dependency. Some approaches achieve automatic parameter detection, but only by exploiting domain-specific knowledge or training data [En18, KT17]. In contrast, our method minimizes user interaction by automatically selecting abstraction parameters in a data-driven manner, leveraging both low-level color and high-level semantic features to produce structured, adaptive abstractions suitable for robotic painting.

Robotic Painting Systems Robotic painting systems range from expressive sketching machines to fully automated brush-based setups. Paul the Robot, developed by Patrick Tresset and collaborators [TL12, TF13], is a widely exhibited robotic arm designed for live sketching, particularly portraits. Its success highlights the artistic and performative possibilities of robotic systems. Other approaches explore alternative materials and mechanisms, such as sponge-based painting or modified brush tools to support diverse physical media styles [SCS*22, KKK*21, KKL*23]. A significant contribution to brush-based robotic painting is the previous iteration of the e-David system, developed at the University of Konstanz by Thomas Lindemeier [LPD13]. The system utilizes an industrial robotic arm to iteratively apply brush strokes using a fixed color palette, guided by a visual feedback loop. After each pass, the canvas is analyzed, and new strokes are selected to reduce local color discrepancies. While this approach achieves visually accurate results, it relies heavily on pixel-wise color matching and lacks structural or semantic abstraction. Consequently, the system tends to overpaint already covered areas and offers limited flexibility for stylistic variation. Recent work has extended the e-David platform toward more structured, region-based painting strategies, including investigations into brush dynamics and hierarchical image representation [GGD18, GPD20, GD22] with Stroh et al. [SGD23] introducing a shape-tree-based method for panoptic abstraction, merging regions by fixed color thresholds. While it allows per-object control, the method requires significant manual tuning and struggles with large homogeneous areas, where minimal bridges of similarly colored pixels connecting distinct regions can lead to excessive merging. Our approach addresses this issue by using semantic saliency in a graph-based merging strategy, thus preserving important substructures and reducing user effort.

3. Method

Figure 2 outlines the steps of our region merging-based image abstraction method, which consists of three main components. First, we begin with a color-based over-segmentation [FH04]. Second, we *generate feature maps* that incorporate machine-learning-based semantic information, depth estimations, and segmentation masks, which are combined with classical edge detector masks to create a saliency weight map. Finally, we utilize an adjacency graph structure to evaluate the similarities between the segmented regions, using their color similarity and values from the saliency map to determine which regions to merge in the *density-based RAG merging* step.

Color-based Over-segmentation As an initialization step, we apply a standard graph-based over-segmentation [FH04] to divide the image into a large set of small, minimally abstracted regions that generally align with local color and luminance edges. This step efficiently produces regions that align with edges in the input image. Such edges are typically detected along local pixel color and luminance differences. However, this approach can be error-prone, for example, when bordering objects are separated by a soft brightness or color gradient, where the color changes gradually across large parts of the input image, and thus, no prominent border can be identified. In natural image scenes, the importance of image edges depends on the visual complexity of an area and the scale and number of features in the image.

To address this issue, we incorporate image semantics into the partitioning process by combining color-based over-segmentation with object-level scene understanding. Specifically, we employ the publicly available panoptic segmentation model OneFormer to identify relevant scene objects within the input image [JLC*23]. While interactive segmentation methods such as Segment Anything could also be applied, they require user-provided prompts to generate object

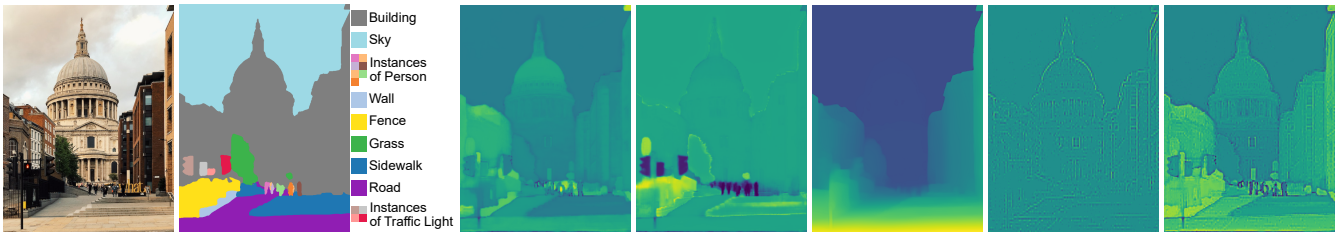


Figure 3: *Semantic feature saliency maps used for panoptic segmentation (see Figure 2): (a) Input image [REN22]; (b) Panoptic segmentation using OneFormer model [JLC*23], (c) Saliency map from the Out-of-Distribution last-layer gradients of the panoptic OneFormer model, computed as proposed by Wang et al. [WJ24], (d) Saliency map obtained using Monte-Carlo Dropout on the panoptic OneFormer model, (e) Depth estimation using the DepthAnything model [YKH*24], (f) Model inference logits of the SegFormer model [Din22] passed through a low-pass filter to extract feature edges, (g) The combined weight map generated by our proposed method*

masks. In contrast, our method applies user control in the subsequent abstraction stage, while the segmentation itself is performed automatically. Therefore, we rely on OneFormer to provide the initial scene segmentation for all presented results [ZYZ*23, LZS*25]. Figure 3 illustrates the panoptic segmentation results for one of the scenes used in our experiments. For the region merging process in the RAG-filtering step of our pipeline, we use high-level semantic object information to merge only regions of the same panoptic class. We use the OneFormer model trained on the COCO dataset [LMB*14] as semantic guidance for the proposed image abstraction method. Figure 3b illustrates the segmentation of Figure 3a in different panoptic classes, represented by different virtual colors.

3.1. Feature Map Generation

A successful image abstraction process requires preserving essential object substructures while avoiding the over-merging of regions from different semantic classes. Panoptic segmentation provides a strong foundation for this task by identifying semantic classes and instances within an image. However, it also introduces challenges, such as erroneously grouping instances of the same class into a single panoptic region or incorrectly identifying substructures (e.g., within buildings, as shown in Figure 3b). Relying only on panoptic segmentation or color differences may result in abstractions failing to capture the complexity and recognizability of the original image. To address these challenges, we introduce a saliency-driven region merging approach. We compute a unified saliency weight map by linearly combining multiple feature detection techniques, allowing us to identify critical regions and object features that should be preserved during abstraction while balancing local context, content importance, and user preference. Regions with high contrast or importance are weighted more heavily, ensuring they are less likely to be merged into unrelated regions. Without this saliency guidance, visually important structures are easily lost, as illustrated in Figure 4d. Applying the unified saliency map (Figure 4c) yields abstractions that better preserve structural details, as shown in Figure 4e. In the remainder of this section, we detail the feature detection techniques used to compute the unified saliency map.

Out-of-Distribution Gradients: Current panoptic segmentation models [JLC*23] rely on predefined datasets (e.g., COCO [LMB*14], Cityscapes [COR*16]), which limits their ability to generalize to unseen objects or substructures. This constraint can result in important image details being overlooked during the abstraction process. We employ an Out-of-Distribution feature detection strategy to address this critical limitation [HG18], which enriches the feature representation and expands the range of detectable objects. We use two different feature detection strategies on the same pre-trained panoptic segmentation model. The first is Monte-Carlo Dropout [GG16], a probabilistic technique that estimates model uncertainty by performing multiple stochastic forward passes through a neural network with dropout enabled. As a second detector, we compute gradients from the normalized logits of the last layer of the panoptic OneFormer segmentation model [JLC*23], as a variation of the class-specific gradient weighting proposed by [WJ24]. This generates smooth gradients by taking the average gradients of Monte-Carlo Dropout input perturbations, instead of injecting noise perturbations on the input. We can calculate both feature detectors without additional overhead or model retraining. Both the dropout uncertainty and the calculated gradients, when treated as saliency maps, highlight regions of the image where the model detects potential features, even if they are not explicitly labeled in the training dataset [GG16, WJ24, STK*17]. Figure 3c and Figure 3d illustrate how this technique enhances feature detection, identifying substructures like facade details and the plateau of stairs in Figure 3a, which are otherwise grouped into a single region by the initial panoptic segmentation. The key advantages of using OoD gradients are:

1. **Dataset Independence**: Unlike standard segmentation logits, OoD gradients are less constrained by the training dataset and can detect novel or unexpected features [LPB17, HG18].
2. **Substructure Detection**: OoD techniques excel at identifying smaller, semantically significant substructures, such as textures and edges within objects [LLS17].
3. **Uncertainty Estimation**: The uncertainty captured by MCD allows us to prioritize regions that are likely to contain meaningful features [GG16, OFR*19].

Integrating these gradients into the saliency map ensures that the abstraction process preserves important features, even when working with diverse and unfamiliar input images.

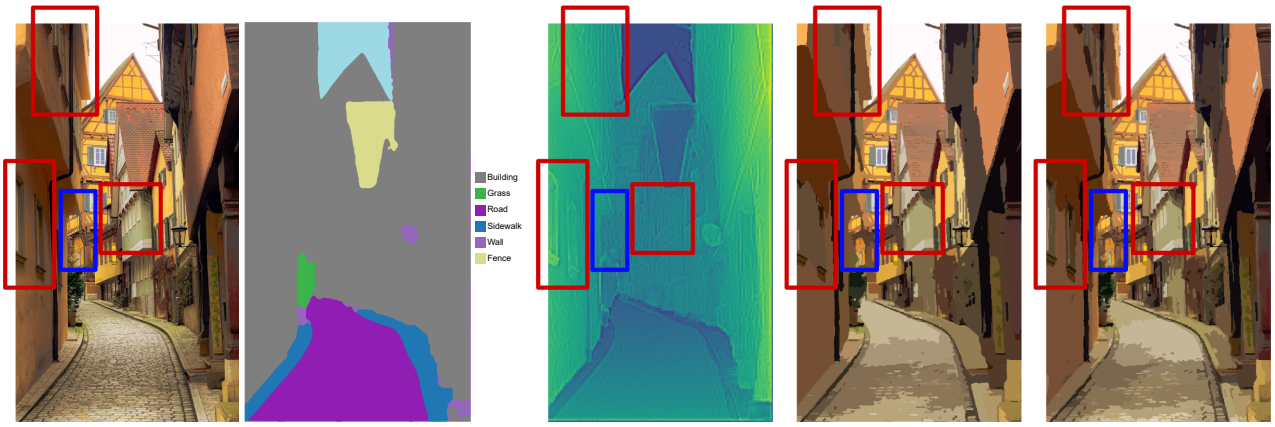


Figure 4: *Influence of Saliency: (a) original image [Tab25]; (b) pantopic segmentation; (c) our generated saliency map; (d) Result without semantic saliency; (e) Result with applying semantic saliency, the object details are much better preserved*

Depth Estimation: Depth provides critical cues about the spatial organization of objects in a scene, improving image segmentation and abstraction by emphasizing depth-based boundaries [SLL23]. We use the "Depth Anything" model [YKH*24] to compute a depth map (Figure 3e), which is then incorporated into the saliency map. Depth cues are particularly valuable for separating overlapping objects and enhancing scene coherence in complex images.

SegFormer segmentation: To capture fine-grained details, we include normalized logits passed through a high-pass filter to extract salient edges from a SegFormer model fine-tuned on face parsing tasks [Din22]. We have found that this component excels in identifying intricate patterns beyond just facial features. In Figure 3f, the model highlights facade details and structures that complement other saliency components.

Saliency Weight Map and User Control: The saliency components are combined into a single weight map using a fixed, weighted linear combination. The combined saliency map is shown in Figure 3g. Users have the additional ability to adjust the abstraction to leverage the strength of the feature detection models. For instance, applying a stronger weight for face segmentation saliency to portrait image inputs or increasing the weight of depth estimation can be particularly useful for indoor scenes, for which depth estimation is especially effective due to the large amount of training data. In some instances, it can also be used to reduce the weight of a particular saliency method that mispredicts large areas or introduces hallucinated features on a particular input image [CMB*20]. This issue mainly occurs when the input images differ substantially from the training data. To address this, users can inspect individual saliency components and deactivate certain detectors as needed.

3.2. Automatic Level-of-Detail Selection

Traditional global abstraction approaches often fail to handle images with high variability in visual complexity. They are prone to merging semantically different regions or missing essential visual features. Therefore, we propose a *locally adaptable image abstraction method* that balances low-level perceptual features, such as color similarity, with high-level semantic information. Our approach automatically selects suitable parameters to preserve critical details while meaningfully simplifying the image. Additionally, it provides the user with fine control over abstraction levels for different objects.

Region Adjacency Graph (RAG) Construction: At the core of our method lies the region adjacency graph (RAG), which represents regions and their spatial relationships throughout the abstraction process. The graph is initialized from an over-segmentation computed by the Felzenszwalb-Huttenlocher algorithm [FH04], where each segment forms a node. Each node stores:

- **Average color**, computed from the region's pixel values.
- **Semantic weight**, obtained by averaging pixel values from the combined saliency map.
- **Panoptic class**, obtained from the panopitc segmentation at the region's location

Edges connect spatially adjacent regions and are assigned weights that quantify pairwise dissimilarity. We define the edge weight w_{ij} between regions i and j as follows:

$$w_{ij} = \frac{\Delta E_{ij}}{\Delta E_{\max}} + \frac{|S_i - S_j|}{S_{\max}},$$

where ΔE_{ij} is the CIE LAB 2000 color difference [LCR01] between regions, S_i and S_j are the semantic saliency values, and the denominators normalize each term. This ensures that the RAG jointly encodes both low-level color similarity and high-level semantic similarity. Throughout the merging process, regions are only merged if they are connected by an edge in the RAG, maintaining spatial coherence and preventing disjointed areas from being combined.

Region Abstraction via RAG Filtering: To control the abstraction process, we apply a two-stage edge filtering procedure. First, edges connecting regions with different panoptic object labels are removed to ensure that only semantically coherent regions may be merged. For each panoptic object, we then identify the corresponding induced subgraph, containing all regions assigned to that object and all edges connecting them in the RAG. Within each induced subgraph, edges

are further removed from the RAG according to their weights: edges with weights exceeding a threshold are removed, disconnecting regions with high dissimilarity in color or semantic saliency. This prevents visually distinct regions from being merged, helping to preserve important internal object structures.

Density-Based RAG Merging: Even after filtering high-weight edges, some visually distinct regions may remain connected through narrow "bridges" — small chains of adjacent regions that are mutually similar in color or semantics, but cumulatively link otherwise distinct areas. These bridging structures often occur along smooth gradients, textures, or thin connecting structures (e.g., vegetation, shadows, or reflections), and can result in undesired merges that span across visually or semantically distinct regions. We illustrate this in Figure 12, where visually distinct regions are merged based on color similarity along a narrow bridge highlighted in red. To address this, we apply the Louvain community detection algorithm [BGLL08] to each filtered subgraph in the RAG. The algorithm partitions the subgraphs into densely connected communities by optimizing the modularity metric:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$

where A_{ij} are the edge weights, k_i is the sum of weights adjacent to node i, m is the sum of all edge weights, and c_i indicates the assigned community. The Kronecker delta function $\delta(c_i, c_j)$ equals 1 if nodes i and j belong to the same community. By incorporating our color- and saliency-based similarity weights into the modularity calculation, regions with stronger similarity are more likely to form communities, while weakly connected regions remain separate. This prevents unintended merges along narrow bridges and leads to structurally meaningful abstraction.

Threshold Selection for Abstraction Control: The edge weight threshold controlling the filtering step directly influences the level of abstraction. To systematically select meaningful thresholds, we analyze how different thresholds affect the resulting number of regions after community detection. For each panoptic object, we compute the cumulative distribution function (CDF) that maps threshold values to the number of remaining regions in a panoptic object. This provides a direct correspondence between the threshold and the abstraction level: higher thresholds allow for more merges, yielding fewer regions. We automatically identify significant thresholds using *knee-point detection* [SAIR11], which locates points of rapid curvature change in the CDF, often marking the transition between meaningful abstraction and over-merging. The automatically determined thresholds can be further refined by user-defined abstraction ranges, providing fine-grained control.

With this, our method automates the selection of abstraction parameters per semantic class while intuitively allowing users to adjust a region's importance with values between 0 (no abstraction) and 1 (maximum). These parameters enable users to narrow the search space for knee-point detection within the cumulative saliency distribution, allowing for fine-tuned control through per-class abstraction control, while reducing manual effort.

To ensure stable knee-point detection for all possible CDFs in our scenario, we introduce a preprocessing step that enforces the convexity assumptions of the algorithm. Monotonous data points are filtered to ensure strict monotonicity, followed by fitting a smooth spline to approximate a convex function. This transformation allows stable knee-point detection even for irregular one-dimensional distributions and is essential for robust threshold selection across diverse input images.

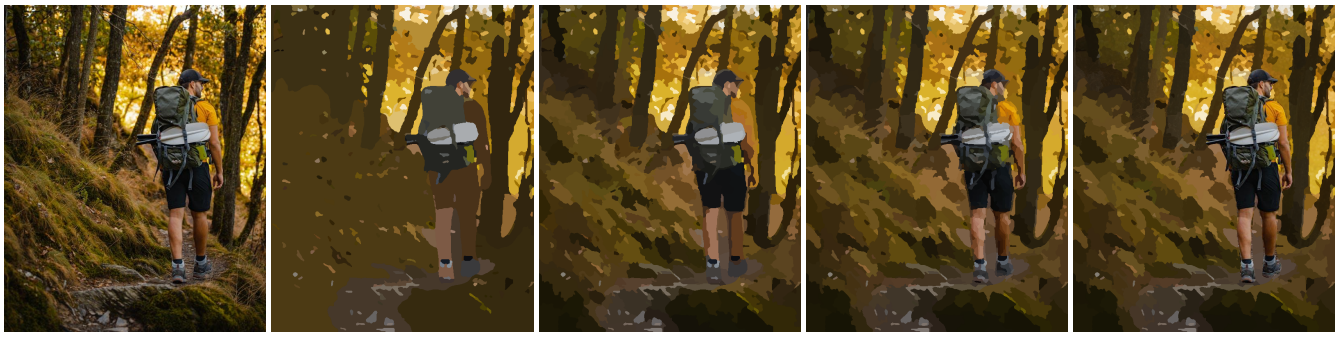
(a) (b)

Figure 5: *Rendering of a street scene: (a) Original image [SIN23]; (b) Result of our method with only 20% of the regions*

The **final abstraction** reflects both the user-defined and automatically determined levels of detail. Prominent features and key boundaries are preserved, while less significant regions are merged to simplify the image in a meaningful way. This adaptive method enables applications in artistic rendering and robotic painting, producing abstractions that strike a balance between simplicity and detail. We map all average region colors to a given color palette for robotic painting. This results in a non-overlapping, region-based image abstraction conforming to a given color palette.

4. Results

We present results demonstrating our abstraction framework's ability to generate structured, semantically-aware image simplifications suitable for robotic painting. Our method selectively abstracts different scene elements based on semantic relevance and visual homogeneity. Background elements, such as roads, skies, or grass, are heavily simplified, while foreground elements, like people, signs, or flowers, retain finer detail. Figures 5 and 6 show this adaptivity in action: signage, windows, and the hiker are preserved with greater detail, while roads, skies, and vegetation are abstracted into larger regions. In Figure 5, the road is heavily abstracted, reducing 4455 over-segmented regions to 829 in total, while structural details like brickwork and tree foliage are preserved. Figure 6 further illustrates semantic adaptivity. While the forest is heavily abstracted, the hiker remains detailed. Varying the Louvain density parameter shows how our method prevents over-merging across thin connections. In the final result (Figure 6d), 7287 initial regions are reduced to 862, with abstraction thresholds computed automatically based on user-specified abstraction bounds (0.05 for the hiker, 0.4 for vegetation).



(a) (b) (c) (d) (e)

Figure 6: *Rendering with emphasis on person hiking in a forest: (a) original image [Pho22]; (b) abstraction without incorporating the Louvain community connectivity, visibly leading to region bridging; (c) abstraction with connectivity considered but with the same level of abstraction for all semantic classes; (d) result of our method with abstracted foliage and detailed hiker; (e) abstracted forest background without any abstraction performed on the hiker in the foreground*

Painting Abstractions with a Robot We use our abstractions as input for a polygon-filling robotic painting pipeline with the e-David painting robot, where each abstracted region is treated as an individual polygon to be filled sequentially. The system computes efficient filling trajectories for each region, optimizing stroke coverage while minimizing unnecessary movements and overlapping brush strokes. During painting, the robot operates in a closed feedback loop, continuously monitoring the painted result and adjusting its actions to compensate for deviations from the expected outcome caused by physical factors such as brush deformation, paint distribution, or surface irregularities. This allows the system to robustly realize the digital abstraction into a physical painting while preserving both structural accuracy and visual quality. Semantic cues inform region boundaries and brush strategies (e.g., stroke orientation or error tolerance near edges). Color quantization using K-means further reduces the necessary pigment preparation.

In physical paintings, strongly simplified regions regain visual richness through the inherent variability introduced by the brush, paint, and texture of the canvas. This natural variation reintroduces subtle details into otherwise flat areas, enhancing depth and visual interest while enabling efficient coverage of large regions during robotic execution. Figure 1 illustrates the full process: From the input image (Figure 1a) through abstraction and color reduction (Figure 1d) to a finished painting (Figure 1e) executed with a DaVinci College 8 brush and water-mixable oil paints. For this result, image processing, including model inferences and abstraction, takes approximately 1 to 2 minutes for an image with dimensions of 640px × 853px on an 11th-generation Intel(R) Core(TM) i9-11900K @ 3.50GHz with 32 GB of RAM and an NVIDIA RTX 3070Ti.

The remaining time (approx. 15 hours) is dedicated to mixing paints, preparing the canvas, and the iterative robotic painting process, which includes automated visual feedback. The stroke planning we used to fill the abstracted areas is based on region contours and the directional gradient field of the image. Strokes are drawn at constant speed, while local stroke directions are adapted during painting using visual feedback to avoid overpainting and ensure proper coverage.

The robot autonomously washes its brush and selects pigment from prepared dishes. However, brush changing is not automated, and a human monitors paint viscosity and pigment availability to ensure consistency during long painting sessions. We selected oil paint for the following main reasons: flexibility, support for layered and wet-in-wet techniques due to the slower drying time than, e.g., acrylics, and color stability after drying–qualities that are beneficial for detailed, iterative robotic painting. Figure 8 and Figure 7 illustrate

(a) (b)

Figure 7: *Robotically painted result: (a) Original image [Uni20]; (b) Painted with the e-David robot system using water-mixable oil paints on a 30cm×40cm canvas board*

painted abstractions of a winter scene and an urban environment, produced with 15 and 17 colors, respectively. In the winter scene, the simplified geometry of the snow-covered house, with softly varying blue tones and warm window lights, conveys both structure and atmosphere. The urban abstraction retains key architectural features such as building facades, rooftops, and foreground elements, while preserving the overall spatial composition. Both paintings remain feasible for robotic execution, with completion times of approximately 12 hours for the winter scene and around 15 hours for the urban scene, while adhering to material and process constraints. In the supplementary material, we showcase a more varied selection of abstracted scenes and additional results.

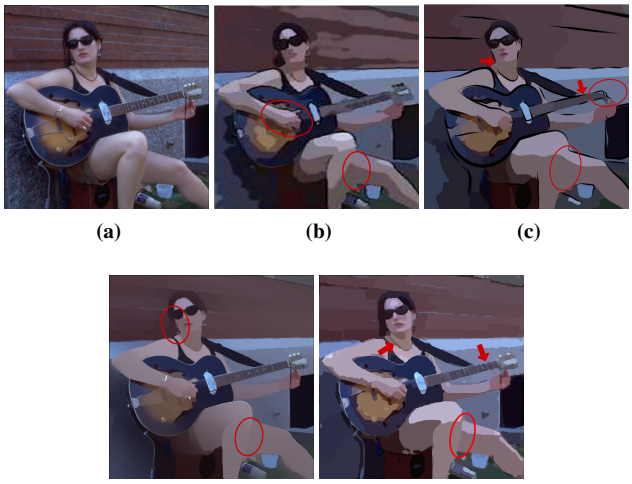
(a) (b)

Figure 8: *Robotically painted result: (a) Original image [Ste24]; (b) Painted with the e-David robot system using water-mixable oil paints on a 30cm×40cm canvas board*

5. Discussion

Our method significantly reduces the number of individual regions in an abstracted image, typically by up to 80% to 90% from the initial over-segmentation, to be within the time and detail constraints of the robotic painting system, while preserving the most relevant image features. As shown in the abstraction results, the combination of color, panoptic segmentation, and semantic saliency effectively resolves object boundaries, particularly where there are no substantial color differences. It prevents the merging of semantically distinct image regions in the RAG. In this manner, we can safely avoid merging different objects and features, provided the models produce good predictions on the input image. Users can iteratively refine the importance values for individual objects by recomputing the abstraction starting from the threshold selection step. This avoids recomputing the initial over-segmentation, the various model inferences, and the RAG construction, thus enabling a quick exploration of possible abstraction results.

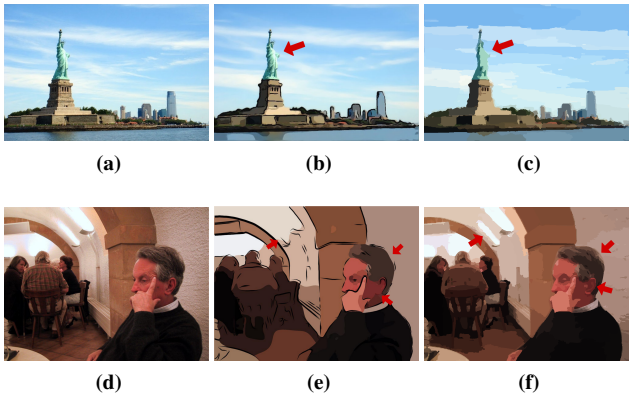
The benefit of incorporating saliency information becomes particularly apparent in cases where large panoptic regions encompass diverse substructures and local contexts. As shown in Figure 4, the facade class covers most of the scene (Figure 4b), containing multiple architectural elements such as windows, doors, and neighboring buildings. Without saliency guidance, these distinct structures are prone to over-merging: in the blue-marked areas, adjacent buildings blend into a single region, while in the red-marked areas, features such as windows are lost entirely (Figure 4c). Panoptic segmentation assigns a uniform label to the entire facade but lacks the granularity to capture such internal variation. The saliency map (Figure 4c) compensates for this limitation by providing additional local importance cues beyond color, helping to preserve semantically meaningful substructures even within large semantic regions. This leads to improved abstractions that retain important details, as demonstrated in the final result (Figure 4e).



(a) (b) (c)

(d) (e)

Figure 9: *Visual abstraction comparison: (a) original image, (b) Kyprianidis et al. [KD08], (c) Sadreazami et al. [SAM17], (d) Bi et al. [BHY15]; and (e) our method*



(a) (b) (c)

(d) (e) (f)

Figure 10: *Visual abstraction comparison: (a,d) original image, (b) Cong et al. [CTD11], (e) DeCarlo et al. [SD02], (c,f) our method*

5.1. Comparison to Related Methods

Different methods for artistic image abstraction aim to achieve diverse goals, utilizing various styles to create different aesthetic impressions. This makes quantitative comparisons between methods in this domain inherently difficult. However, a qualitative comparison is still possible. We aim to create a framework for image abstraction that incorporates object-level user guidance and automatic parameter selection, enabling the creation of image regions inspired by human region-based painting styles. Figures 9, 10, 11 and 12 show results of other image abstraction methods alongside our abstractions for a qualitative comparison.

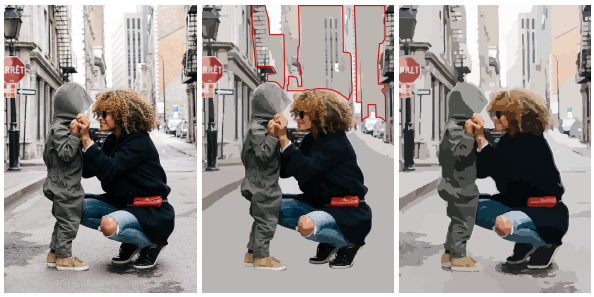
Using object-based adjustable abstraction parameter, it is possible to give greater visual fidelity to the woman with her guitar in Figure 9d while still maintaining a strong level of abstraction in the background wall. The regions are abstracted without relying on a strong smoothing strategy. Thus, our result retains small details such

(a) (b) (c) (d)

Figure 11: *Method Comparison: (a) original image [Pat18]; (b) result using the shape-tree abstraction framework [SGD23] (2526 shapes); (c) rendered result using LIVE [MZX*22] with 610 overlapping paths; (d) our method (634 shapes)*

as the woman's necklace and the fret lines of the guitar, which are lost in the other methods shown in Figure 9b and Figure 9c. With the influence of the introduced saliency weight map, her two legs in Figure 9a can be prevented from merging, unlike other abstraction frameworks. The results by Cong et al. [CTD11] in Figure 10b preserve the sky and the Statue of Liberty almost exactly as it is in the original, creating a visual disparity between the abstract and photorealistic regions.

In contrast, our method can handle such complex image structures and find a suitable region composition to abstract all semantic areas. Work by DeCarlo et al. [SD02] as shown in Figure 10e utilizes gaze detection to enhance local abstraction fidelity to highlight important features, separating pixels into foreground and background. They merge salient features in areas without much focus, and only slightly abstract the area that the user focuses on. Our method selects suitable abstraction levels for all semantic objects in the image, allowing it to produce more varied results for different scene elements.



(a) (b) (c)

Figure 12: *Method Comparison: (a) original image [DS15]; (b) result using the shape-tree abstraction framework [SGD23] (3312 shapes); (c) our method (794 shapes).*

5.2. Comparison to robotic abstraction method

Figure 11 illustrates that our method can create more detailed abstraction results without producing over-segmented abstractions when compared to previous work by Stroh et al. [SGD23]. By limiting the effects of bridging regions in our framework, we avoid large merges that occur within the tree crown. In Figure 11b, the sidewalk and street are reduced to just one shape, which removes the curb and foliage on the ground, as these regions were merged through bridging regions. Our method significantly enhances the recognizability of the abstracted object, allowing for high levels of abstraction without removing substantial parts of the objects. Similarly, in Figure 12, our method prevents the merging of large background regions with buildings, thereby preserving the scene composition. In Figure 12b, the image parts highlighted in red spanning multiple buildings and the sky are merged into large homogenous regions, while our proposed method in Figure 12c retains those background features. Thus, we can create abstractions with higher visual fidelity while reducing the quantitative number of regions.

We note that the recent method for robotic painting proposed by [SGD23] requires extensive parameter tuning to balance abstraction and detail, as these parameters have to be manually adjusted for each object in the scene. In contrast, our approach automatically determines all individual parameters within a predefined abstraction range, which remains consistent across all objects in the demonstration. This automation significantly reduces the need for manual tuning, streamlining the abstraction process while maintaining high visual fidelity.

5.3. Comparison to Optimization-based Vectorization methods

As an additional application, our region-based abstraction method can be used to vectorize images. Given a pixel image, it can be transformed into vector representations (SVGs) by partitioning it into semantic objects using our proposed framework. This is similar to work on photograph ClipArt generation by Favreau et al. [FLB17]. However, while other vectorization methods in this domain can produce region-based abstractions with high visual fidelity, they are unsuited for robotic painting due to their reliance on overlapping regions [MZX*22, LLGRK20]. These methods often create visually appealing results by defining regions with significant overlap, which

(a) (b) (c)

Figure 13: *Automatically generated hierarchy of abstraction levels.*

works well in digital renderings but poses practical difficulties for physical realization. When applied to robotic painting, overlapping regions require painting multiple layers on top of each other, which can lead to unintended color mixing effects. Such effects are often undesirable unless used explicitly as a stylistic device; more specifically, they can compromise the clarity and precision of the final painting, as essential features overlap or become fragmented.

Furthermore, vectorization methods, because of their global optimization processes tend to be computationally demanding and require significant time to converge [MZX*22, LLGRK20]. For instance, the result produced by the LIVE framework [MZX*22] in Figure 11c took approximately an hour to compute, whereas our method, shown in Figure 11d, generates results in under two minutes, including over-segmentation and saliency mask inference. This reduction in processing time enables a more interactive workflow, allowing users to iteratively refine abstraction levels. Additionally, our approach provides localized control over abstraction, enabling users to balance detail and simplification more effectively.

5.4. Limitations and Future Work

Unlike a human artist who can develop a comprehensive plan for an entire painting and thoughtfully place or position elements within the scene, our approach is bound to the arrangement of the input image itself. It only abstracts individual regions in the input image without a scene understanding of a human artist, as they, e.g., do not just underrepresent uninteresting regions. They might replace complex or detailed background objects with simpler shapes or change the composition of objects in these regions to be more aesthetically pleasing to the artist's subjective opinion. The proposed system cannot change the object composition or the geometric representation of a complex object, similar to work by Kang et al. [KL08] that can simplify object geometry.

While the presented robotic paintings were created using a single abstraction layer, the proposed method inherently supports the construction of hierarchical abstraction levels with varying degrees of detail similar to [FXDG17]. Such a layered approach would enable multi-stage painting strategies, similar to how a human painter conceptualizes a painting. Coarse abstractions serve as underpaintings for subsequent, more detailed layers, potentially enhancing depth, structure, and visual contrast in the final artwork. Importantly, once the region adjacency graph and cumulative distribution functions have been precomputed, the system can efficiently generate a full hierarchy of abstraction levels by simply varying the range of the automatically selected parameters. This process requires no further human interaction and can be controlled by interpolating the abstraction range values, allowing for the rapid generation of multiple abstraction layers at different levels of detail. Figure 13 illustrates this capability, showing how progressively finer abstractions are generated from the same input data without reprocessing. The abstraction ranges from the colored panoptic segmentation (left) to progressively finer abstractions (right). Although this capability was not used for the current painting results, it is supported by the framework and will be explored in the future.

This work was developed as part of a broader interdisciplinary project in close collaboration with professional artists and painters from the EACVA † project. Their ongoing feedback significantly influenced both the design of the abstraction method and the practical considerations of the robotic painting process. In the future, we plan to conduct a formal user study involving both artists and general viewers to assess the painted results. Further, as our abstraction pipeline is media-agnostic, it is possible to apply our method to other media and tools, such as palette knife painting [BSSG20], for example. However, the stroke planning would need to be adapted to suit the physical properties of the chosen tools and materials. This might be an exciting direction to explore in future work.

6. Conclusions

We presented a user-guided image abstraction framework tailored for robotic painting and showcased results painted by the e-David robot. The system integrates panoptic segmentation, semantic feature detection, and color-based over-segmentation. Using a region adjacency graph (RAG) with edges encoding saliency, color, and structural features, we apply automatic knee-point thresholding and Louvain community detection to preserve object boundaries and substructures. This controlled merging process supports structured representations critical for robotic stroke planning, color application, and stylization, making the resulting abstractions well-suited for robotic painting, vectorization, NPR, and artistic rendering. Future work will focus on advanced stroke placement, exploring layered painting approaches, adapting to alternative media, and evaluating the system's expressive potential through user studies with artists and broader audiences.

Acknowledgements

This work was funded by the EACVA (Embodied Agents in Contemporary Visual Art) Project, led by Goldsmiths (UKRI/AHRC grant AH/X002241/1) and the University of Konstanz (grant 508324734, Deutsche Forschungsgemeinschaft/DFG). Patrick Paetzold was funded by DFG Project 251654672 TRR 161 "Quantitative methods for visual computing". Open Access funding enabled and organized by Projekt DEAL.

References

[ASS*10] ACHANTA R., SHAJI A., SMITH K., LUCCHI A., FUA P., SÜSSTRUNK S.: *SLIC superpixels*. Tech. rep., EPFL 149300, 2010. 2

† www.eacva.org, www.edavid.de

[BGLL08] BLONDEL V. D., GUILLAUME J.-L., LAMBIOTTE R., LEFEBVRE E.: Fast unfolding of communities in large networks. *Jour. of statistical mechanics: theory and experiment*, 10 (2008), P10008. doi:10.1088/1742-5468/2008/10/P10008. 6

[BHY15] BI S., HAN X., YU Y.: An L1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. *ACM Trans. Graph. 34*, 4 (July 2015). doi:10.1145/2766946. 8

[BSSG20] BELTRAMELLO A., SCALERA L., SERIANI S., GALLINA P.: Artistic robotic painting using the palette knife technique. *Robotics 9*, 1 (2020), 15. doi:10.3390/robotics9010015. 10

[CLH*16] CHENG M.-M., LIU Y., HOU Q., BIAN J., TORR P., HU S.-M., TU Z.: HFS: hierarchical feature selection for efficient image segmentation. In *Computer Vision – ECCV 2016* (2016), Springer Int. Publishing, pp. 867–882. doi:10.1007/978-3-319-46487-9_53. 2

[CMB*20] CERMELLI F., MANCINI M., BULO S. R., RICCI E., CAPUTO B.: Modeling the background for incremental learning in semantic segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 9230–9239. doi:10.1109/CVPR42600.2020.00925. 5

[COR*16] CORDTS M., OMRAN M., RAMOS S., REHFELD T., ENZWEILER M., BENENSON R., FRANKE U., ROTH S., SCHIELE B.: The Cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition* (2016), pp. 3213–3223. doi:10.1109/CVPR.2016.350. 4

[CTD11] CONG L., TONG R., DONG J.: Selective image abstraction. *The Visual Computer 27* (2011), 187–198. doi:10.1007/s00371-010-0522-2. 8, 9

[Din22] DINU J.: Fine-tuned segformer model for face parsing, 2022. [Online; accessed September 22, 2024]. URL: https://huggingface.co/jonathandinu/face-parsing. 4, 5

[DS15] DE-SILVA S.: Women and child standing on a road, 2015. [Online; accessed February 28, 2023], Unsplash License. URL: https://unsplash.com/de/fotos/httxBNGKapo. 9

[En18] EL-NAGGAR A. M.: Determination of optimum segmentation parameter values for extracting building from remote sensing images. *Alexandria Engineering Journal 57*, 4 (2018), 3089–3097. doi:10.1016/j.aej.2018.10.001. 3

[FH04] FELZENSZWALB P. F., HUTTENLOCHER D. P.: Efficient graph-based image segmentation. *Int. Journ. of Computer Vision 59*, 2 (2004), 167–181. doi:10.1023/B:VISI.0000022288.19776.77. 2, 3, 5

[FLB17] FAVREAU J.-D., LAFARGE F., BOUSSEAU A.: Photo2ClipArt: Image abstraction and vectorization using layered linear gradients. *ACM Trans. Graph. 36*, 6 (2017), 1–11. doi:10.1145/3130800.3130888. 9

[FXDG17] FARAJ N., XIA G.-S., DELON J., GOUSSEAU Y.: A generic framework for the structured abstraction of images. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering* (2017), NPAR, Association for Computing Machinery. doi:10.1145/3092919.3092930. 2, 10

[GD22] GÜLZOW J. M., DEUSSEN O.: Region-based approaches in robotic painting. *Arts 11*, 4 (2022). doi:10.3390/arts11040077. 1, 3

[GG01] GOOCH B., GOOCH A.: *Non-photorealistic rendering*. A K Peters/CRC Press, 2001. doi:10.1201/9781439864173. 1

[GG16] GAL Y., GHAHRAMANI Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning* (2016), vol. 48, PMLR, pp. 1050–1059. URL: https://proceedings.mlr.press/v48/gal16.html. 4

[GGD18] GÜLZOW J. M., GRAYVER L., DEUSSEN O.: Self-improving robotic brushstroke replication. *Arts 7*, 4 (2018). doi:10.3390/arts7040084. 3

[GPD20] GÜLZOW J. M., PAETZOLD P., DEUSSEN O.: Recent developments regarding painting robots for research in automatic painting, artificial creativity, and machine learning. *Applied Sciences 10*, 10 (2020). doi:10.3390/app10103396. 2, 3

[Her10] HERTZMANN A.: Non-photorealistic rendering and the science of art. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering* (2010), NPAR, Association for Computing Machinery, pp. 147–157. doi:10.1145/1809939.1809957. 1

[HG18] HENDRYCKS D., GIMPEL K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks, 2018. doi:10.48550/arXiv.1610.02136. 4

[HJA20] HO J., JAIN A., ABBEEL P.: Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (2020), NeurIPS, Curran Associates Inc. 2

[HJA24] HIRSCHORN O., JEVNISEK A., AVIDAN S.: Optimize & reduce: A top-down approach for image vectorization. *Proc. of the AAAI Conference on Artificial Intelligence 38*, 3 (Mar. 2024), 2148–2156. doi:10.1609/aaai.v38i3.27987. 3

[HP10] HE L., PAPPAS T. N.: An adaptive clustering and chrominance-based merging approach for image segmentation and abstraction. In *2010 IEEE Int. Conf. on Image Processing* (2010), pp. 241–244. doi:10.1109/ICIP.2010.5651905. 2

[JLC*23] JAIN J., LI J., CHIU M., HASSANI A., ORLOV N., SHI H.: Oneformer: One transformer to rule universal image segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 2989–2998. doi:10.1109/CVPR52729.2023.00292. 2, 3, 4

[KCWI13] KYPRIANIDIS J. E., COLLOMOSSE J., WANG T., ISENBERG T.: State of the "art": A taxonomy of artistic stylization techniques for images and video. *IEEE Trans. on Visualization and Computer Graphics 19*, 5 (2013), 866–885. doi:10.1109/TVCG.2012.160. 1

[KD08] KYPRIANIDIS J. E., DÖLLNER J.: Image abstraction by structure adaptive filtering. In *TPCG* (2008), The Eurographics Association, pp. 51–58. doi:10.2312/LocalChapterEvents/TPCG/TPCG08/051-058. 8

[KHG*19] KIRILLOV A., HE K., GIRSHICK R., ROTHER C., DOLLÁR P.: Panoptic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), pp. 9396–9405. doi:10.1109/CVPR.2019.00963. 2

[KKK*21] KARIMOV A., KOPETS E., KOLEV G., LEONOV S., SCALERA L., BUTUSOV D.: Image preprocessing for artistic robotic painting. *Inventions 6*, 1 (2021). doi:10.3390/inventions6010019. 3

[KKL*23] KARIMOV A., KOPETS E., LEONOV S., SCALERA L., BUTUSOV D.: A robot for artistic painting in authentic colors. *Journal of Intelligent & Robotic Systems 107*, 3 (Mar 2023), 34. doi:10.1007/s10846-023-01831-4. 3

[KL08] KANG H., LEE S.: Shape-simplifying image abstraction. *Computer Graphics Forum 27*, 7 (2008), 1773–1780. doi:10.1111/j.1467-8659.2008.01322.x. 10

[KT17] KAVZOGLU T., TONBUL H.: Selecting optimal SLIC superpixels parameters by using discrepancy measures. 38th Asian Conference on Remote Sensing (ACRS). URL: https://acrs-aars.org/proceeding/ACRS2017/ID_5_749/191.pdf. 3

[Lab24] LABS B. F.: Flux. https://github.com/black-forest-labs/flux, 2024. 2

[LCR01] LUO M. R., CUI G., RIGG B.: The development of the cie 2000 colour-difference formula: Ciede2000. *Color Research & Application 26*, 5 (2001), 340–350. doi:10.1002/col.1049. 5

[LLGRK20] LI T.-M., LUKÁČ M., GHARBI M., RAGAN-KELLEY J.: Differentiable vector graphics rasterization for editing and learning. *ACM Trans. Graph. 39*, 6 (Nov. 2020). doi:10.1145/3414685.3417871. 2, 9, 10

[LLS17] LIANG S., LI Y., SRIKANT R.: Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690* (2017). doi:10.48550/arXiv.1706.02690. 4

[LMB*14] LIN T.-Y., MAIRE M., BELONGIE S., HAYS J., PERONA P., RAMANAN D., DOLLÁR P., ZITNICK C. L.: Microsoft COCO: Common objects in context. In *Computer Vision – ECCV* (2014), Springer International Publishing, pp. 740–755. doi:10.1007/978-3-319-10602-1_48. 4

[LPB17] LAKSHMINARAYANAN B., PRITZEL A., BLUNDELL C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (2017), vol. 30 of *NIPS'17*, Curran Associates, Inc. 4

[LPD13] LINDEMEIER T., PIRK S., DEUSSEN O.: Image stylization with a painting machine using semantic hints. *Computers & Graphics 37*, 5 (2013), 293–301. doi:10.1016/j.cag.2013.01.005. 1, 2, 3

[LZS*25] LI F., ZHANG H., SUN P., ZOU X., LIU S., LI C., YANG J., ZHANG L., GAO J.: Segment and recognize anything at any granularity. In *Computer Vision – ECCV 2024* (2025), Springer Nature Switzerland, pp. 467–484. doi:10.1007/978-3-031-73195-2_27. 4

[Mou12] MOULD D.: Region-based abstraction. In *Image and Video-Based Artistic Stylisation*. Springer, 2012, pp. 125–147. doi:10.1007/978-1-4471-4519-6. 2

[MZX*22] MA X., ZHOU Y., XU X., SUN B., FILEV V., ORLOV N., FU Y., SHI H.: Towards layer-wise image vectorization. In *Proc. of the IEEE Conf. on computer vision and pattern recognition* (2022). doi:10.1109/CVPR52688.2022.01583. 2, 9, 10

[OFR*19] OVADIA Y., FERTIG E., REN J., NADO Z., SCULLEY D., NOWOZIN S., DILLON J., LAKSHMINARAYANAN B., SNOEK J.: Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems* (2019), vol. 32, Curran Associates, Inc. 4

[Pat18] PATEL R.: Cars paraked in line next to an orange tree, 2018. [Online; accessed January 11, 2023], Unsplash License. URL: https://unsplash.com/es/fotos/yK_mJlzLQUY. 9

[Pho22] PHOTOS T.: Hiker on forest trail, 2022. [Online; accessed September 27, 2024], Pexels License. URL: https://www.pexels.com/photo/a-man-hiking-a-forest-13622369/. 7

[PV08] PENG B., VEKSLER O.: Parameter selection for graph cut based image segmentation. In *Proceedings of the British Machine Vision Conference* (2008), BMVA Press, pp. 16.1–16.10. doi:10.5244/C.22.16. 3

[RC13] ROSIN P., COLLOMOSSE J. (Eds.): *Image and video-based artistic stylisation*. Springer London, 2013. doi:10.1007/978-1-4471-4519-6. 1

[REN22] REN Y.: Red tree, 2022. [Online; accessed August 20, 2023], Pexels License. URL: https://www.pexels.com/photo/woman-sitting-on-a-wall-by-the-river-14597109/. 4

[RPG*21] RAMESH A., PAVLOV M., GOH G., GRAY S., VOSS C., RADFORD A., CHEN M., SUTSKEVER I.: Zero-shot text-to-image generation. doi:10.48550/arXiv.2102.12092. 2

[SAIR11] SATOPAA V., ALBRECHT J., IRWIN D., RAGHAVAN B.: Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In *31st Int. Conf. on distributed computing systems workshops* (2011), pp. 166–171. doi:10.1109/ICDCSW.2011.20. 6

[SAM17] SADREAZAMI H., ASIF A., MOHAMMADI A.: Iterative graph-based filtering for image abstraction and stylization. *IEEE Trans. on Circuits and Systems II: Express Briefs 65*, 2 (2017), 251–255. doi:10.1109/TCSII.2017.2669866. 2, 8

[SCS*22] SCALERA L., CANEVER G., SERIANI S., GASPARETTO A., GALLINA P.: Robotic sponge and watercolor painting based on image-processing and contour-filling algorithms. *Actuators 11*, 2 (2022). doi:10.3390/act11020062. 3

[SD02] SANTELLA A., DECARLO D.: Abstracted painterly renderings using eye-tracking data. In *Proc of the 2nd Int. Symp. on Non-Photorealistic Animation and Rendering* (2002), ACM, pp. 75–82 + 159. doi:10.1145/508530.508544. 8, 9

[SGD23] STROH M., GÜLZOW J.-M., DEUSSEN O.: Semantic image abstraction using panoptic segmentation for robotic painting. In *Proceedings of Vision, Modeling, and Visualization* (2023), The Eurographics Association. doi:10.2312/vmv.20231235. 2, 3, 9

[SIN23] SINGH R.: Tree, 2023. [Online; accessed June 6, 2024], Pexels License. URL: https://www.pexels.com/de-de/foto/hauser-strasse-hotel-burgersteig-16433679/. 6

[SLL23] SCHÖN R., LUDWIG K., LIENHART R.: Impact of pseudo depth on open world object segmentation with minimal user guidance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2023), pp. 4809–4819. doi:10.1109/CVPRW59228.2023.00509. 5

[SMO23] SCHALDENBRAND P., MCCANN J., OH J.: Frida: A collaborative robot painter with a differentiable, real2sim2real planning environment. In *IEEE International Conference on Robotics and Automation (ICRA)* (2023), pp. 11712–11718. doi:10.1109/ICRA48891.2023.10160702. 2

[Ste24] STEIN M.: House, 2024. 8

[STK*17] SMILKOV D., THORAT N., KIM B., VIÉGAS F., WATTENBERG M.: SmoothGrad: removing noise by adding noise. doi:10.48550/arXiv.1706.03825. 4

[Tab25] TABIKH O.: Charming medieval street in quaint european village, 2025. Online; accessed June 3, 2025. URL: https://www.pexels.com/photo/charming-medieval_street-in-quaint-european-village-31316339/. 5

[TF13] TRESSET P., FOL LEYMARIE F.: Portrait drawing by Paul the robot. *Computers & Graphics 37*, 5 (2013), 348–363. doi:10.1016/j.cag.2013.01.012. 2, 3

[TL12] TRESSET P. A., LEYMARIE F. F.: Sketches by Paul the robot. In *Proc. of the Eighth Annual Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging* (2012), The Eurographics Association, p. 17–24. doi:10.2312/COMPAESTH/COMPAESTH12/017-024. 2, 3

[Uni20] UNIVERSITY OF KONSTAZ: Luftaufnahmen Universität Konstanz, 2020. Photos taken by Patrick Doodt. 7

[WJ24] WANG H., JI Q.: Epistemic uncertainty quantification for pretrained neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), pp. 11052–11061. doi:10.1109/CVPR52733.2024.01051. 2, 4

[XCGN16] XU Y., CARLINET E., GÉRAUD T., NAJMAN L.: Hierarchical segmentation using tree-based shape spaces. *IEEE trans. on pattern analysis and machine intelligence 39*, 3 (2016), 457–469. doi:10.1109/TPAMI.2016.2554550. 2

[YKH*24] YANG L., KANG B., HUANG Z., XU X., FENG J., ZHAO H.: Depth anything: Unleashing the power of large-scale unlabeled data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), pp. 10371–10381. doi:10.1109/CVPR52733.2024.00987. 2, 4, 5

[ZWQL16] ZHOU C., WU D., QIN W., LIU C.: An efficient two-stage region merging method for interactive image segmentation. *Computers & Electrical Engineering 54* (2016), 220–229. doi:10.1016/j.compeleceng.2015.09.013. 2

[ZYZ*23] ZOU X., YANG J., ZHANG H., LI F., LI L., WANG J., WANG L., GAO J., LEE Y. J.: Segment everything everywhere all at once. In *Proceedings of the 37th International Conference on Neural Information Processing Systems* (2023), NeurIPS, Curran Associates Inc. 4

[ZZL24] ZHANG P., ZHAO N., LIAO J.: Text-to-vector generation with neural path representation. *ACM Trans. Graph. 43*, 4 (July 2024). doi:10.1145/3658204. 3