Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 23, no. 3, pp. 525–552 (2019)
DOI: 10.7155/jgaa.00501

Aesthetic Discrimination of Graph Layouts

Tamara Mchedlidze[1] *Alexey Pak*[2] *Moritz Klammler*[1]

[1]Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
[2]Fraunhofer Institute of Optronics, System Technologies and Image Exploitation, Fraunhoferstraße 1, 76131 Karlsruhe, Germany

Abstract

This paper addresses the following basic question: given two layouts of the same graph, which one is more aesthetically pleasing? We propose a neural network-based discriminator model trained on a labeled dataset that decides which of two layouts has a higher aesthetic quality. The feature vectors used as inputs to the model are based on known graph drawing quality metrics, classical statistics, information-theoretical quantities, and two-point statistics inspired by methods of condensed matter physics. The large corpus of layout pairs used for training and testing is constructed using force-directed drawing algorithms and the layouts that naturally stem from the process of graph generation. It is further extended using data augmentation techniques. Our model demonstrates a mean prediction accuracy of 97.58 %, outperforming discriminators based on stress and on the linear combination of popular quality metrics by a margin of 2 to 3%.

The present paper extends our contribution to the Proceedings of the 26th International Symposium on Graph Drawing and Network Visualization (GD 2018) and is based on a significantly larger dataset.

Submitted:	Reviewed:	Revised:	Accepted:
November 2018	January 2019	May 2019	June 2019
	Final:	Published:	
	July 2019	September 2019	

Article type:	Communicated by:
Regular paper	T. Biedl and A. Kerren

E-mail addresses: mched@iti.uka.de (Tamara Mchedlidze) alexey.pak@iosb.fraunhofer.de (Alexey Pak) moritz.klammler@alumni.kit.edu (Moritz Klammler)

1 Introduction

What makes a drawing of a graph aesthetically pleasing? This admittedly vague question is central to the field of Graph Drawing which has over its history suggested numerous answers. Borrowing ideas from Mathematics, Physics, Arts, etc., many researchers have tried to formalize the elusive concept of aesthetics.

In particular, dozens of formulas collectively known as *drawing aesthetics* (or, more precisely, *quality metrics* [11]) have been proposed that attempt to capture in a single number how beautiful, readable and clear a drawing of an abstract graph is. Of those, simple metrics such as the number of edge crossings, minimum crossing angle, vertex distribution or angular resolution parameters are obviously incapable *per se* of providing the ultimate aesthetic statement. Advanced metrics may represent, for example, the energy of a corresponding system of physical bodies [10, 14]. This approach underlies many popular graph drawing algorithms [48] and often leads to pleasing results in practice. However, it is known that low values of energy or stress do not always correspond to the highest degree of symmetry [54] which is an important aesthetic criterion [40].

Another direction of research aims to narrow the scope of the original question to specific application domains, focusing on the purpose of a drawing or possible user actions it may facilitate (*tasks*). The target parameters – readability and the clarity of representation – may be assessed via user performance studies. However, even in this case such aesthetic notions as symmetry still remain important [40]. In general, aesthetically pleasing designs are known to positively affect the apparent and the actual usability [33, 50] of interfaces and induce positive mental states of users, enhancing their problem-solving abilities [13].

In this work, we offer an alternative perspective on the aesthetics of graph drawings. First, we address a slightly modified question: "Of two given drawings of the same graph, which one is more aesthetically pleasing?". With that, we implicitly admit that "the ultimate" quality metric may not exist and one can hope for at most a (partial) ordering of layouts with respect to their aesthetic value. Instead of a metric, we therefore search for a binary *discriminator function* of graph drawings. As limited as it is, it could be useful for practical applications such as picking the best answer out of the outputs of several drawing algorithms or resolving local minima in layout optimization.

Second, similarly to several known works [7, 19] we believe that by combining multiple metrics computed for each drawing one has a better chance of capturing complex aesthetic properties. We thus also consider a "meta-algorithm" that aggregates several "input" metrics into a single value. However, unlike the recipes in the above papers, we do not specify the form of this combination *a priori* but let an artificial neural network "learn" it based on a sample of labeled training data. In the recent years, machine learning techniques have proven useful in such aesthetics-related tasks as assessing the appeal of 3D shapes [8] or cropping photos [32]. Our network architecture is based on a so-called *Siamese neural network* [4] – a generic model specifically designed for binary functions of same-kind inputs.

Finally, we acknowledge that any simple or complex input metric may become crucial to the answer in some cases that are hard to predict *a priori*. We therefore implement as many input metrics as we can and relegate their ranking to the model. In addition to those known from the literature, we implement a few novel metrics inspired by statistical tools used in Condensed Matter Physics and Crystallography, which we expect to be helpful in capturing the symmetry, balance, and salient structures in larger graphs. These metrics are based on so-called *syndromes* – variable-size multi-sets of numbers computed for a graph or its drawing (e.g. vertex coordinates or pairwise distances). In order to reduce these heterogeneous multi-sets to a fixed-size *feature vector* (input to the discriminator model), we perform a *feature extraction* process which may involve steps such as creating histograms or performing regressions.

Due to the lack of human-labeled data and the practical complexities of conducting a large-scale user study to collect subjective aesthetic preferences, we have performed our experiments based on artificially labelled data. First, we have compiled a large corpus of publicly available and generated graphs. Second, in order to produce layout pairs with known aesthetical ordering, we have exploited several strong assumptions. In particular, we postulate that layouts produced by force-directed algorithms and those naturally stemming from the generation process are aesthetically pleasing to a high degree, and that random disturbances introduced to such layouts always reduce their aesthetic quality. These assumptions are indirectly supported by user studies stating that people prefer layouts with fewer crossings [21, 39], and generally value medium levels of complexity [2] and high level of symmetry [51].

In our experiments, our discriminator model outperforms the known (metric-based) algorithms and achieves an average accuracy of 97.58 % when identifying the "better" graph drawing out of a pair. The project source code including the data generation procedure is available online [27].

The remainder of this paper is structured as follows. In section 2 we briefly overview the state-of-the-art in quantifying graph layout aesthetics. Section 4 discusses the used syndromes of aesthetic quality, section 5 – the feature extraction, and section 6 – the discriminator model. The dataset used in our experiments is described in section 7. The results and the comparisons with the known metrics are presented in section 8. Section 9 summarises the paper and provides an outlook for the future work.

2 Related Work

According to empirical studies, graph drawings that maximize one or several quality metrics are more aethetically pleasing and easier to read [18, 19, 36, 41, 53]. For instance, in their seminal work, Purchase et al. have established [40] that higher numbers of edge crossings and bends as well as lower levels of symmetry negatively influence user performance in graph reading tasks.

Many graph drawing algorithms attempt to optimize multiple quality metrics at once. As one way to combine them, Coleman and Parker [7] have used a weighted sum of "simple" metrics. Huang et al. [19] have additionally considered the effects of the interactions between "simple" metrics (see [37] and [22]), and extra terms accounting for possible measurement errors.

In another work, Huang et al. [23] have empirically demonstrated that their "aggregate" metric is sensitive to quality changes and is correlated with the human performance in graph comprehension tasks. They have also noticed that the dependence of aesthetic quality on input quality metrics can be non-linear (e.g. a quadratic relationship better describes the interplay between crossing angles and drawing quality [20]). Our work extends this idea as we allow for arbitrary non-linear dependencies implemented by an artificial neural network.

Purchase et al. [39] have studied subjective preferences of users for UML diagrams. One of the conclusions of that study was that users prefer diagrams with fewer crossings. Huang et al. have collected subjective usability ratings of sociograms [21] and established that drawings with fewer crossings are favored. Purchase et al. [41] have asked users to select the most "liked" drawing out of drawings with straight-line and circular edges, with the results indicating the preference for circular drawings. Carbon et al. [5] have altered the outlines of graph visualization and asked for beauty ratings. The drawings that looked "curvier" were preferred by the users. Further, in cognitive psychology it is generally known that a medium level of complexity gains the highest aesthetic value (see, e.g., [2]). It is also known that high levels of symmetry ease the way we perceive objects [51] and information visualizations [52].

In evolutionary graph drawing approaches, several techniques have been suggested to "train" a *fitness function*[*] from the user's responses as a composition of several known quality metrics. Masui [31] modeled the fitness function as a linear combination in which the weights are obtained via genetic programming from the pairs of "good" and "bad" layouts provided by users. The so-called co-evolution was used by Barbosa and Barreto [1] to evolve the weights of the fitness function in parallel with a drawing population in order to match the ranking made by users. Spönemann and others [46] suggested two alternative techniques. In the first one, the user directly chooses the weights with a slider. In the second, they select good layouts from the current population and the weights are adjusted according to the selection. Rosete-Suarez [42] determined the relative importance of individual quality metrics based on user inputs. Several machine learning-based approaches to graph drawing are described by dos Santos Vieira et al. [9]. Recently, Kwon et al. [30] presented a novel work on topological similarity of graphs. Their goal was to avoid expensive computations of graph layouts and their quality measures. The resulting system was able to sketch a graph in different layouts and estimate corresponding quality metrics.

In most cases above, the evaluated layout is represented as an abstract 2D object with precisely known parameters such as vertex coordinates (this paradigm is also assumed in the current paper). All involved aesthetically-related metrics and syndromes are therefore related to the graph structure and geometrical relations such as distances and angles. In an alternative approach recently sug-

[*] Objective function in genetic algorithms that summarizes optimization goals.

gested by Haleem et al. [17], quality metrics are predicted by an artificial neural network given a rasterized image of a graph layout over some fixed-resolution pixel grid. On the one hand, using raw pixel values allows one to avoid (sometimes costly) "true" computations of edge crossing, node spread, node overlap, etc and simplifies the feature extraction. On the other hand, any chosen resolution is obviously sub-optimal for too large or too small graphs, or for graph layouts that lead to "inconvenient" rasterizations. It is, however, possible, that some hybrid approach that exploits both the abstract layout geometry as well as its images (possibly with adaptive or multi-scale resolution) may eventually lead to solid improvements over "pure" methods.

3 Definitions

In this paper we consider general simple graphs $G = (V, E)$ where $V = V(G)$ and $E = E(G)$ are the vertex and edge sets of G with $|V| = n$ and $|E| = m$. A *drawing* or *layout* of a graph is its graphical representation where vertices are drawn as points or small circles, and the edges as straight line segments. Vertex positions in a drawing are denoted by $\vec{p}^{\,k} = (p_1^k, p_2^k)^{\mathrm{T}}$ for $k = 1, \ldots, n$ and their set $P = \{\vec{p}^{\,k}\}_{k=1}^{n}$. Further, we use $\mathrm{dist}_G(u, v)$ to denote the *graph-theoretical distance* – the length of the shortest path between vertices u and v in G – and $\mathrm{dist}_\Gamma(u, v)$ for the Euclidean distance between u and v in the drawing $\Gamma(G)$.

4 Quality Syndromes of Graph Layouts

A *quality syndrome* of a layout Γ is a multi-set of numbers that share a common interpretation, and that are known or suspected to correlate with the aesthetic quality (e.g., all pairwise angles between incident edges in Γ). In this section we describe several syndromes that are implemented in our code and that have been inspired by popular quality metrics and common statistical tools.

When developing these definitions, we assumed that respective algorithms must remain practical for graphs containing up to a few thousand of nodes. At this scale, it is often still possible for humans to evaluate graphs visually and many popular layout algorithms converge relatively fast. On the other hand, such objects may already be studied using statistical distributions. Of course, if a given application typically requires much larger or much smaller graphs, the syndromes should be respectively adjusted.

The following list is by no means exhaustive, nor do we claim syndromes below to be necessary or independent. Our subsequent discriminator model accepts any combination of syndromes; their choices remain to be systematically investigated.

`PRINVEC1` and `PRINVEC2` The two principal axes of the set P. If we define a covariance matrix $C = \{c_{ij}\}$, $c_{ij} = \frac{1}{n}\sum_{k=1}^{n}(p_i^k - \overline{p_i})(p_j^k - \overline{p_j})$, $i, j \in \{1, 2\}$, where $\overline{p_i} = \frac{1}{n}\sum_{k=1}^{n} p_i^k$ are the mean values over each dimension, then `PRINVEC1` and `PRINVEC2` will be its eigenvectors.

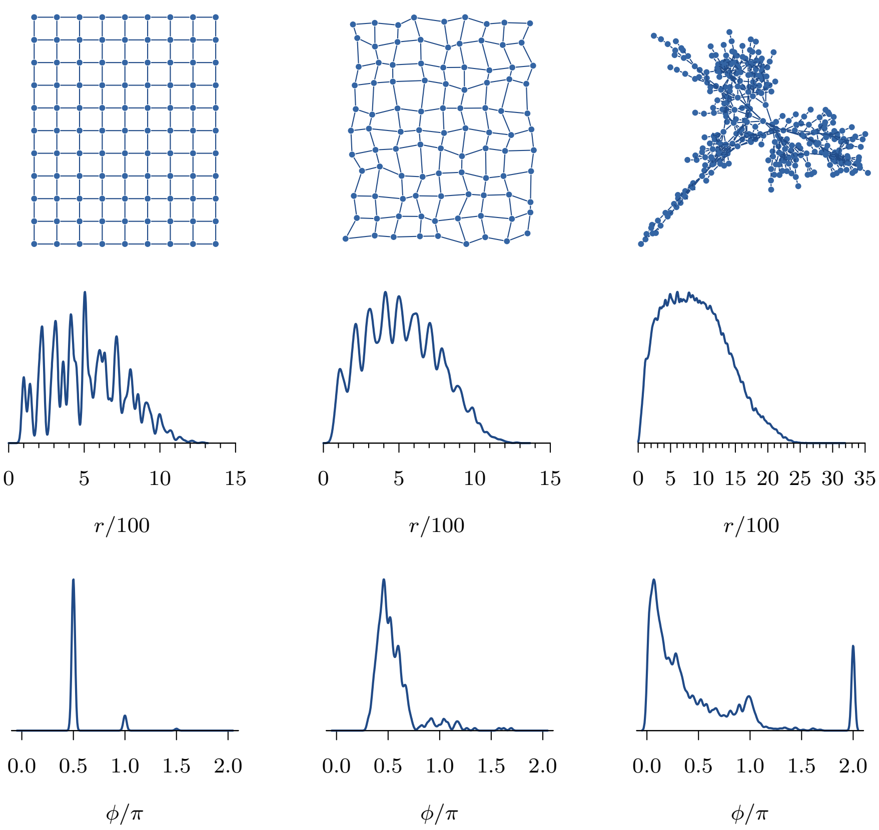


Figure 1: Illustration of the syndromes RDF_GLOBAL and ANGULAR. Upper row, from left to right: proper, distorted layouts of a regular grid, and a force-directed layout of an irregular graph ("power grid"). Central row: smoothed relative frequency distributions for the RDF_GLOBAL syndromes computed for the respective layouts in the upper row. The isolated peaks in the leftmost distribution correspond to characteristic distances in the lattice. In the central plot, these peaks are widened due to random distortion. In the rightmost plot, no regular structure can be identified. Lower row: smoothed relative frequency distributions for the ANGULAR syndromes. The leftmost plot clearly shows the dominance of angles proportional to $\pi/2$. In the rightmost plot, the distinctive peak at $\phi = 2\pi$ corresponds to the large number of degree one vertices.

`PRINCOMP1` **and** `PRINCOMP2` Projections of vertex positions onto $\vec{v}_1 = \texttt{PRINVEC1}$ and $\vec{v}_2 = \texttt{PRINVEC2}$, that is, $\{\langle(\vec{p}^j - \overline{\vec{p}}), \vec{v}_i\rangle\}_{j=1}^n$ for $i \in \{1, 2\}$ where $\langle\cdot,\cdot\rangle$ denotes the scalar product.

`ANGULAR` Let $A(v)$ denote the sequence of edges incident to a vertex v, appearing in a clockwise order around it in Γ. Let $\alpha(e_i, e_j)$ denote the clockwise angle between edges e_i and e_j incident to the same vertex. This syndrome is then defined as $\bigcup_{v\in V(G)}\{\alpha(e_i, e_j) : e_i, e_j \text{ are consecutive in } A(v)\}$. Fig. 1 shows histograms of `ANGULAR` for some graphs and layouts. Note that more regular drawings feature better-isolated peaks in the respective histograms.

`EDGE_LENGTH` $\bigcup_{(u,v)\in E(G)}\{\operatorname{dist}_\Gamma(u, v)\}$ is the set of edge lengths in Γ.

`RDF_GLOBAL` $\bigcup_{u\neq v\in V(G)}\{\operatorname{dist}_\Gamma(u, v)\}$ contains distances between all vertices in the drawing. The concept of a *radial distribution function (RDF)* [12] (the distribution of `RDF_GLOBAL`) is borrowed from Statistical Physics and Crystallography and characterizes the regularity of molecular structures. In large graph layouts it captures regular, periodic and symmetric patterns in the vertex positions. Fig. 1 illustrates histograms of `RDF_GLOBAL`. Again, more regular drawings result in better-isolated peaks in the histograms.

`RDF_LOCAL`(d) $\bigcup_{u\neq v\in V(G)}\{\operatorname{dist}_\Gamma(u, v) : \operatorname{dist}(u, v) \leq d\}$ is the set of distances between vertices such that the graph-theoretical distance between them is bounded by $d \in \mathbb{N}$. In our implementation, we compute `RDF_LOCAL`(2^i) for $i \in \{0, \ldots, \lceil\log_2(D)\rceil\}$ where D is the diameter of G. `RDF_LOCAL`(d) in a sense interpolates between `EDGE_LENGTH` $(d = 1)$ and `RDF_GLOBAL` $(d \to \infty)$.

`TENSION` $\bigcup_{u\neq v\in V(G)}\{\operatorname{dist}_\Gamma(u, v)/\operatorname{dist}_G(u, v)\}$ are the ratios of Euclidean and graph-theoretical distances computed for all vertex pairs. `TENSION` is motivated by and is related to the well-known stress function [24].

Note that before computing the quality syndromes, we *normalize* all layouts so that the center of gravity of V is at the origin and the mean edge length is fixed in order to remove the effects of scaling and translation (but not rotation).

5 Feature Vectors

The sizes of quality syndromes are in general graph- and layout-dependent. A neural network, however, requires a fixed-size input. A collection of syndromes is condensed to this *feature vector* via *feature extraction.* Our approach to this step relies on several auxiliary definitions. Let $S = \{x_i\}_{i=1}^p$ be a syndrome with p entries. By S^μ we denote the arithmetic mean and by S^ρ the root mean square of S. We also define a *histogram sequence* $S^\beta = \frac{1}{p}(S_1, \ldots, S_\beta)$ – a vector of normalized counts in a histogram built over S with β bins. The *entropy* [45] of S^β is defined as

$$\mathscr{E}(S^\beta) = -\sum_{i=1}^{p} \log_2(S_i) S_i \,. \tag{1}$$

We expect the entropy, as a measure of disorder, to be related to the aesthetic quality of a layout and convey important information to the discriminator.

The entropy $\mathscr{E}(S^\beta)$ is sensitive to the number of bins β (cf. Fig. 2). In order to avoid biases due to arbitrary choices of β, we compute the entropy for $\beta = 8, 16, \ldots, 512$. After that, we perform a linear regression of $\mathscr{E}(S^\beta)$ as a function of $\log_2(\beta)$. Specifically, we find S^η and S^σ such that $\sum_\beta (S^\sigma \log_2 \beta + S^\eta - \mathscr{E}(S^\beta))^2$ is minimized. The parameters (intercept S^η and slope S^σ) of this regression no longer depend on the histogram size and may be used as feature vector components. Fig. 2 illustrates that the dependence of $\mathscr{E}(S^\beta)$ on $\log_2(\beta)$ is indeed often close to linear and the regression provides a decent approximation.

A histogram over S can be generalized to a continuous *sliding average*

$$S^F(x) = \frac{\sum_{i=1}^{p} F(x, x_i)}{\int_{-\infty}^{+\infty} \mathrm{d}y \ \sum_{i=1}^{p} F(y, x_i)} \,. \tag{2}$$

A natural choice for the kernel $F(x, y)$ is the Gaussian $F_\sigma(x, y) = \exp\left(-\frac{(x-y)^2}{2\sigma^2}\right)$. By analogy to Eq. 1, we may now define the *differential entropy* [45] as

$$\mathscr{D}(S^{F_\sigma}) = -\int_{-\infty}^{+\infty} \mathrm{d}x \ \log_2(S^{F_\sigma}(x))\, S^{F_\sigma}(x) \,. \tag{3}$$

This entropy via kernel function still depends on the filter width σ. Computing $\mathscr{D}(S^{F_\sigma})$ for multiple values of σ as we do for $\mathscr{E}(S^\beta)$ is too expensive. Instead, we have found that using Scott's Normal Reference Rule [44] as a heuristic to fix σ yields satisfactory results, and allows us to define $S^\varepsilon = \mathscr{D}(S^{F_\sigma})$.

Using these definitions, for the most complex syndrome $\texttt{RDF\_LOCAL}(d)$ we introduce $\texttt{RDF\_LOCAL}$ – a 30-tuple containing the arithmetic mean, root mean square and the differential entropy of $\texttt{RDF\_LOCAL}(2^i)$ for $i \in (0, \ldots, 9)$. With that[†], $\texttt{RDF\_LOCAL} = \left(\texttt{RDF\_LOCAL}(2^i)^\mu, \texttt{RDF\_LOCAL}(2^i)^\rho, \texttt{RDF\_LOCAL}(2^i)^\varepsilon\right)_{i=0}^{9}$.

Finally, we assemble the 57-dimensional[‡] feature vector for a layout Γ as

$$F_{\text{layout}}(\Gamma) = \texttt{PRINVEC1} \cup \texttt{PRINVEC2} \cup \texttt{RDF\_LOCAL} \cup \bigcup_S \left(S^\mu, S^\rho, S^\eta, S^\sigma\right) \,,$$

where S ranges over PRINCOMP1, PRINCOMP2, ANGULAR, EDGE_LENGTH, RDF_GLOBAL and TENSION.

In addition, the discriminator model receives the trivial properties of the underlying graph as the second 2-dimensional vector $F_{\text{graph}}(G) = (\log(n), \log(m))$.

[†] Values $i < 10$ are sufficient as no graph in our dataset has a diameter exceeding 2^9.

[‡] The size is one less than expected from the explanation above because we do not include the arithmetic mean for EDGE_LENGTH as it is constant (due to the layout normalization mentioned earlier) and therefore is non-informative.

6 Discriminator Model

Feature extractors such as those introduced in the previous section reduce an arbitrary graph G and its arbitrary layout Γ to fixed-size vectors $F_{\mathrm{graph}}(G)$ and $F_{\mathrm{layout}}(\Gamma)$. Given a graph G and a pair of its alternative layouts Γ_a and Γ_b, the discriminator function DM receives the feature vectors $\vec{v}_a = F_{\mathrm{layout}}(\Gamma_a)$, $\vec{v}_b = F_{\mathrm{layout}}(\Gamma_b)$ and $\vec{v}_G = F_{\mathrm{graph}}(G)$ and outputs a scalar value

$$t = \mathrm{DM}(\vec{v}_G, \vec{v}_a, \vec{v}_b) \in [-1, 1]\,. \tag{4}$$

The interpretation is as follows: if $t < 0$, then the model believes that Γ_a is "prettier" than Γ_b; if $t > 0$, then it prefers Γ_b. Its magnitude $|t|$ encodes the confidence level of the decision (the higher $|t|$, the more solid the answer).

For the implementation of the function DM we have chosen a practically convenient and flexible model structure known as *Siamese neural networks*, originally proposed by Bromley and others [4] that is defined as

$$\mathrm{DM}(\vec{v}_G, \vec{v}_a, \vec{v}_b) = \mathrm{GM}(\vec{\sigma}_a - \vec{\sigma}_b, \vec{v}_G) \tag{5}$$

where $\vec{\sigma}_a = \mathrm{SM}(\vec{v}_a)$ and $\vec{\sigma}_b = \mathrm{SM}(\vec{v}_b)$. The *shared model* SM and the *global model* GM are implemented as multi-layer neural networks with a simple structure shown in Fig. 3. The network was implemented using the *Keras* [25] framework with the *TensorFlow* [49] library as the back-end.

The SM network (Fig. 3(a)) consists of two "dense" (fully-connected) layers, each preceded by a "dropout" layer (discarding 50 % and 25 % of the signals, respectively). Dropout is a stochastic regularization technique intended to avoid overfitting that was first proposed by Srivastava and others [47].

In the GM network (Fig. 3(b)), the graph-related feature vector $\vec{v}_G$ is passed through an auxiliary dense layer, and concatenated with the difference signal $(\vec{\sigma}_a - \vec{\sigma}_b)$ obtained from the output vectors of SM for the two layouts. The final dense layer produces the scalar output value. The first and the auxiliary layers use linear activation functions, the hidden layer uses ReLU [16] and the final layer hyperbolic tangent activation. Following standard practice, the inputs to the network are normalized by subtracting the mean and dividing by the standard deviation of the feature computed over the complete dataset.

As in the case of the syndrome selection, the choices above are not claimed to be optimal. The network structure was developed to provide the maximum flexibility (offered by hidden layers with the full connectivity to the input and the output layers), and, at the same time, to maintain the number of trainable parameters significantly below the number of the available training data entries (to avoid overfitting). Activation function choices roughly represent our assumptions about the variation of the respective signals: feature vector components are generally assumed to be unbounded, while the output value has a well-defined limited range. Similarly, ReLU-activation in the hidden layers is a popular and efficient way to introduce non-linearity in the model. Overall, the present network is a typical and straightforward solution for a situation when the training dataset is small. Should the data availability improve, larger and more complex networks could be considered.

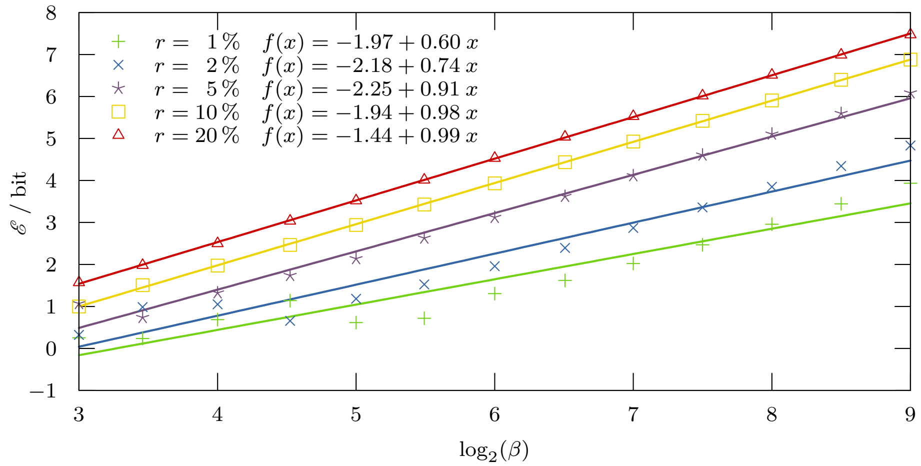


Figure 2: Entropy $\mathcal{E} = \mathcal{E}(S^{\beta})$ computed for histogram sequences S^{β} defined for different numbers of histogram bins β. Different markers (colors) correspond to several layouts of a regular grid-like graph, progressively distorted according to the parameter r. See Fig. 9 for the examples of distorted grid layout. The dependence of $\mathcal{E}$ on $\log_2(\beta)$ is well approximated by a linear function. Both intercept and slope show a strong correlation with the levels of distortion r.

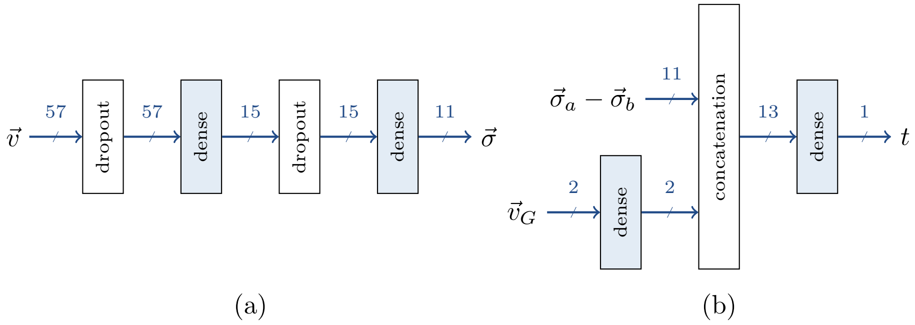


Figure 3: Structure of the neural networks $\mathrm{SM}(\vec{v})$ (a) and $\mathrm{GM}(\vec{\sigma}_a - \vec{\sigma}_b, \vec{v}_G)$ (b). Shaded blocks denote standard network layers, and the numbers on the arrows denote the dimensionality of the respective representations.

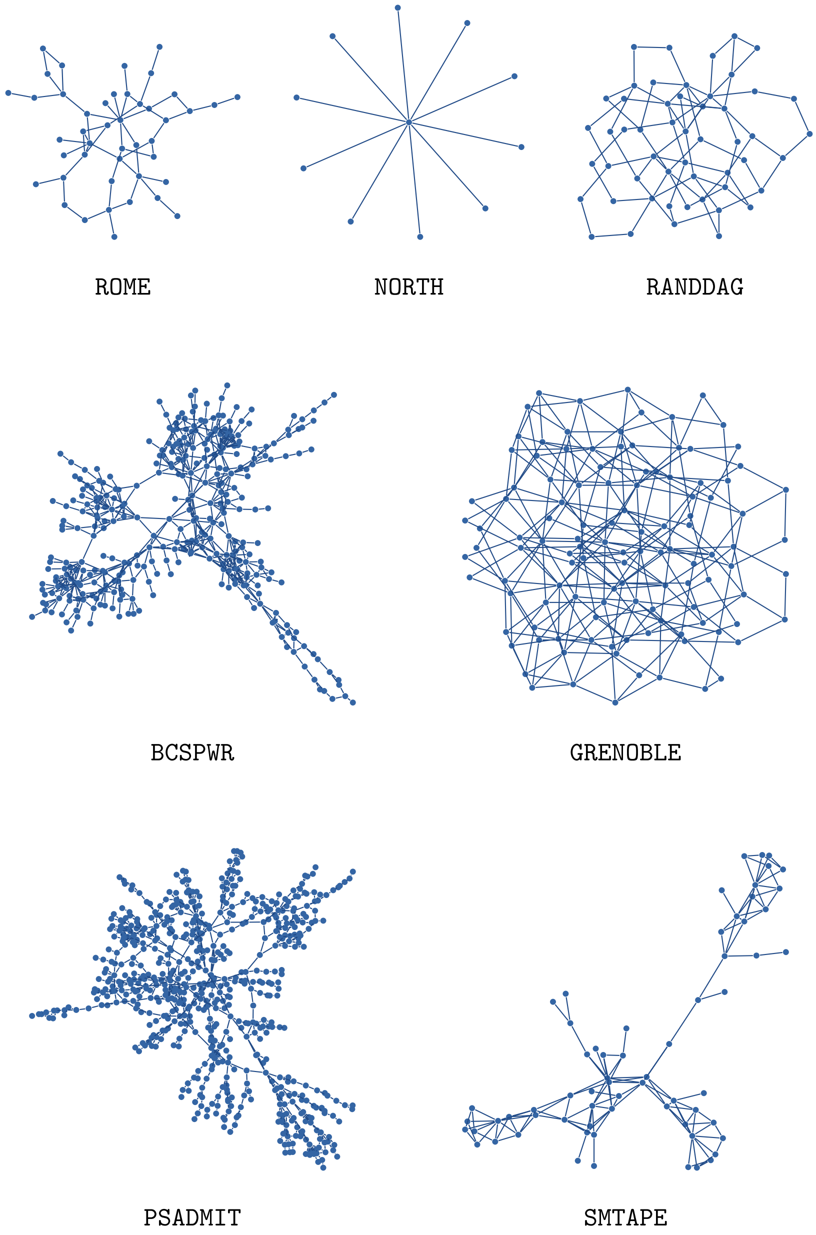


Figure 4: Examples of imported graphs. The BCSPWR, GRENOBLE, PSADMIT and SMTAPE graphs come from the respective datasets in the Harwell-Boeing collection in NIST's "Matrix Market" [3]. All graphs are visualized using the FM^3 algorithm.

In total, the DM model has 1 066 free parameters, trained via stochastic gradient descent-based optimization of the mean squared error (MSE) loss function.

7 Training and Testing Data

For training, all machine learning methods require datasets representing the variability of possible inputs. Our DM model needs a dataset containing graphs, their layouts, and known aesthetic orderings of layout pairs. We have assembled such a dataset using two types of sources. First, we used the collections of the well-known graph archives `ROME`, `NORTH` and `RANDDAG` which are published on `graphdrawing.org` as well as the NIST's "Matrix Market" [3]. See Fig. 4 for examples.

Second, we have generated random graphs using the algorithms listed below. As a by-product, some of them produce layouts that stem naturally from the generation logic. We refer to these as *native* layouts (see [26] for details). Sample graphs with native layouts (where available) are shown in Fig. 5. In experiments, we have limited the graph sizes by 1000 nodes.

`GRID` Regular $n \times m$ grids. Native layouts: regular rectangular grids.

`TORUS1` Same as `GRID`, but the first and the last "rows" are connected to form a 1-torus (a cylinder). No native layouts.

`TORUS2` Same as `TORUS1`, but also the first and the last "columns" are connected to form a 2-torus (a doughnut). No native layouts.

`LINDENMAYER` Uses a stochastic L-system [35] to derive increasingly complex graphs by performing random replacements of individual vertices with more complicated substructures such as an n-ring or an n-clique. Fig. 6 shows all the implemented replacement rules. Produces a planar native layout.

`QUASI`$\langle n \rangle$`D` **for** $n \in \{3, \ldots, 6\}$ Projection of a primitive cubic lattice in an n-dimensional space onto a 2-dimensional plane intersecting that space at a random angle. The native layout follows from the construction.

`MOSAIC1` Starts with a regular polygon and randomly divides faces according to a set of simple rules until the desired graph size is reached. The rules include adding a vertex connected to all vertices of the face; subdividing each edge and adding a vertex that connects to each subdivision vertex; subdividing each edge and connecting them to a cycle. These operations are visualized in Fig. 7. The native layout follows from the construction.

`MOSAIC2` Applies a randomly chosen rule of `MOSAIC1` to every face, with the goal of obtaining more symmetric graphs.

BOTTLE Constructs a graph as a three-dimensional mesh over a random solid of revolution. The native layout is the axonometric projection of the mesh.

For each graph, we have computed force-directed layouts using the FM^3 [15] and stress-minimization [24] algorithms, implemented in the OGDF library [6]. We assume these and native layouts to be generally aesthetically pleasing and call them all *proper* layouts of a graph.

Furthermore, we have generated *a priori* un-pleasing (*garbage*) layouts as follows. Given a graph $G = (V, E)$, we generate a random graph $G' = (V', E')$ with $|V'| = |V|$ and $|E'| = |E|$ and compute a force-directed layout for G'. The coordinates found for the vertices V' are then assigned to V. We call these "phantom" layouts due to the use of a "phantom" graph G'. We find that phantom layouts look less artificial than purely random layouts when vertex positions are sampled from a uniform or a normal distribution. This might be due to the fact that G and G' have the same density and share some beneficial aspects of the force-directed method (such as mutual repelling of nodes). See Fig. 8 for the examples of regular and garbage layouts.

For training and testing of the discriminator model we need a corpus of labeled pairs – triplets (Γ_a, Γ_b, t) where Γ_a and Γ_a are two different layouts for the same graph and $t \in [-1, 1]$ is a value indicating the relative aesthetic quality of Γ_a and Γ_b. A negative (positive) value for t expresses that the quality of Γ_a is superior (inferior) compared to Γ_b and the magnitude of t expresses the confidence of this prediction. We only use pairs with $|t| > 0.05$.

As manually-labelled data were unavailable, we have fixed the values of t as follows. First, we paired a proper and a garbage layout of a graph. The assumption is that the former is always more pleasing (i.e. $t = \pm 1$). Second, in order to obtain more nuanced layout pairs and to increase the amount of data, we have employed the well-known technique of *data augmentation* as follows.

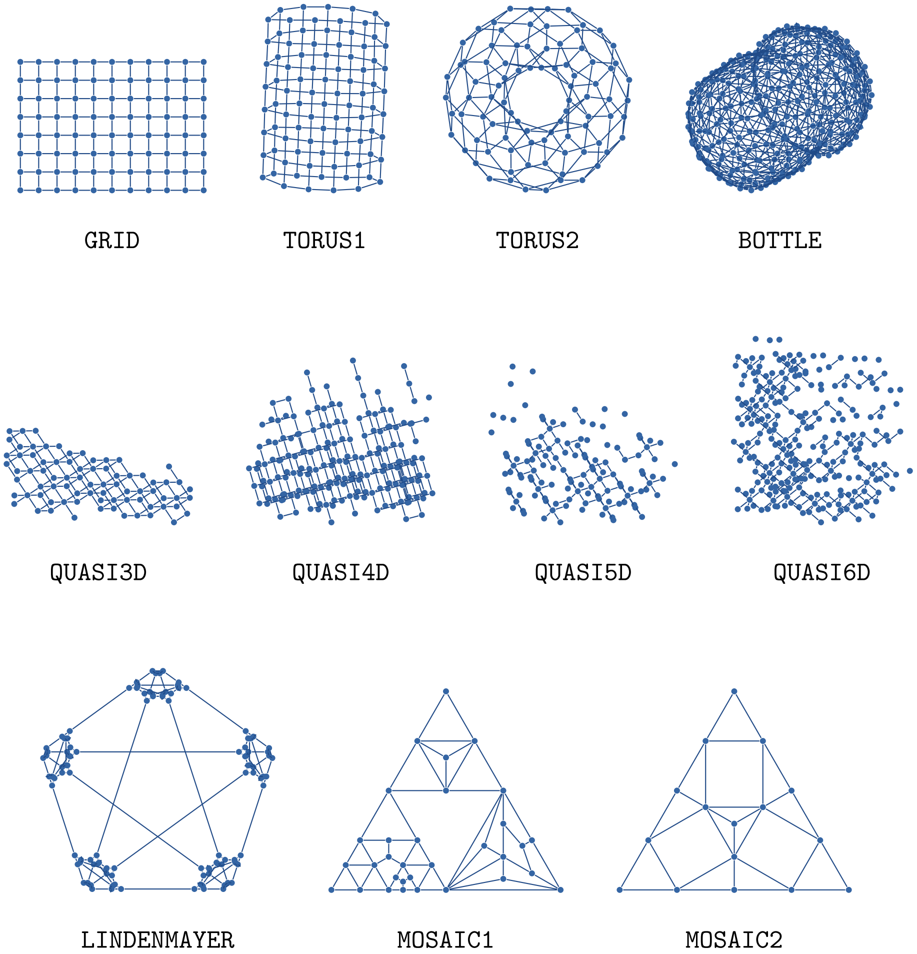
Layout Worsening: Given a proper layout Γ, we apply a transformation designed to gradually reduce its aesthetic quality that is modulated by some parameter $r \in [0, 1]$, resulting in a transformed layout Γ'_r. By varying the degree r of the distortion, we may generate a sequence of layouts ordered by their anticipated aesthetic value: a layout with less distortion is expected to be more pleasing than a layout with more distortion when starting from a presumably decent layout. We have implemented the following worsening techniques. Illustrations of these worsening algorithms can be found in Fig. 9.

PERTURB adds Gaussian noise to each node's coordinates.

FLIP_NODES swaps coordinates of randomly selected node pairs.

FLIP_EDGES same as FLIP_NODES but restricted to connected node pairs.

MOVLSQ applies an affine deformation based on moving least squares suggested (although for a different purpose) by Schaefer et al. [43]. In essence, all vertices are shifted according to some smoothly varying coordinate mapping.

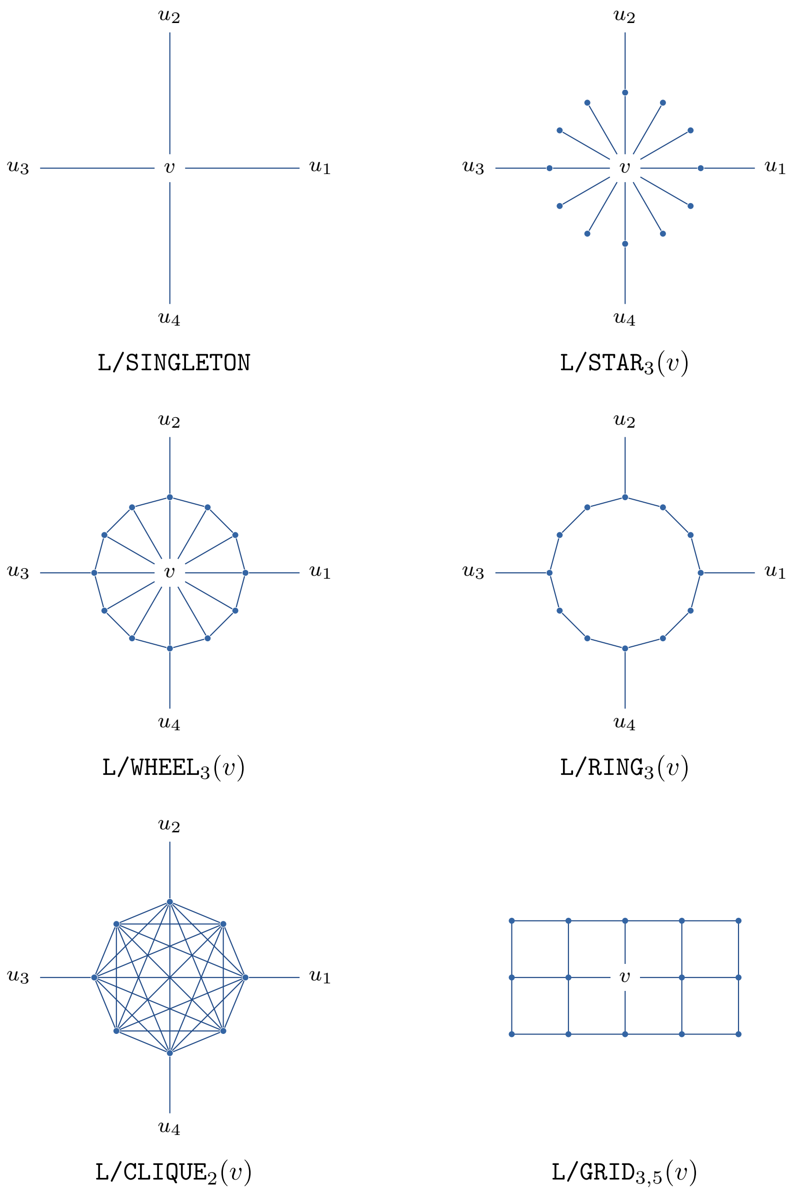


GRID TORUS1 TORUS2 BOTTLE

QUASI3D QUASI4D QUASI5D QUASI6D

LINDENMAYER MOSAIC1 MOSAIC2

Figure 5: Examples of generated graphs labeled by the respective generators. `GRID`, `LINDENMAYER`, `QUASI`$\langle$`n`$\rangle$`D`, `MOSAIC1`, `MOSAIC2` and `BOTTLE` layouts are native. `TORUS1` and `TORUS2` are visualized with the stress-minimization algorithm.



L/SINGLETON

L/STAR$_3(v)$

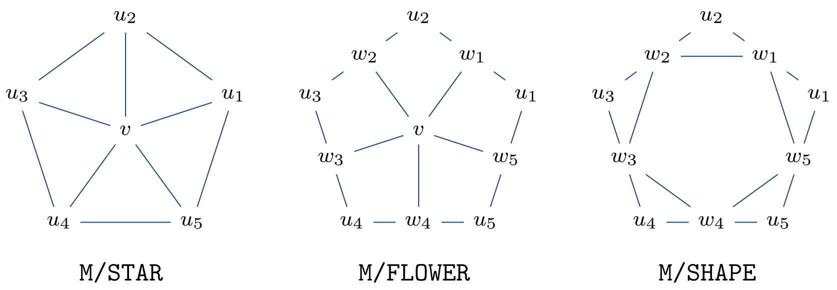
L/WHEEL$_3(v)$

L/RING$_3(v)$

L/CLIQUE$_2(v)$

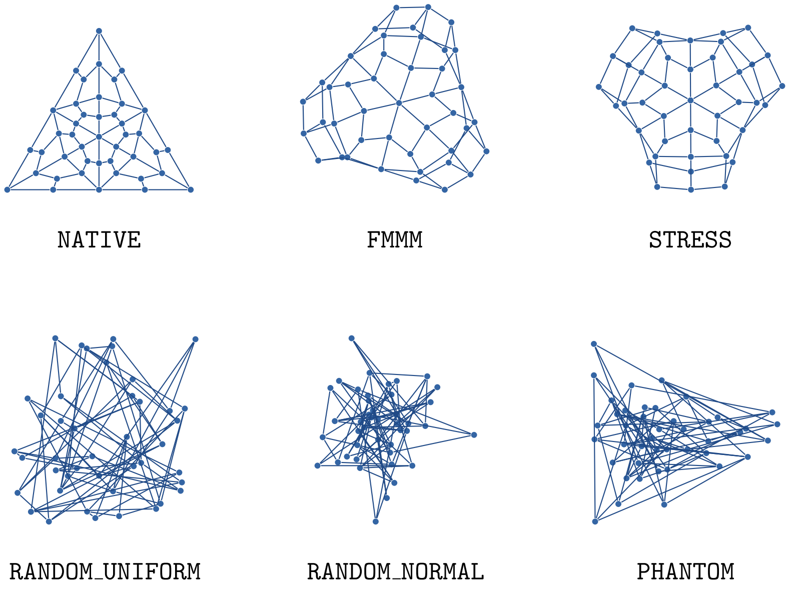
L/GRID$_{3,5}(v)$

Figure 6: Illustration of the LINDENMAYER generator operations. A degree four vertex may be replaced by any of the above subgraphs, except for the bottom right subgraph which replaces a degree zero vertex.



M/STAR M/FLOWER M/SHAPE

Figure 7: Operations of the MOSAIC generator on a pentagonal facet $\{u_1, \ldots, u_5\}$.



NATIVE FMMM STRESS

RANDOM_UNIFORM RANDOM_NORMAL PHANTOM

Figure 8: Examples of different layouts for the same graph. RANDOM_UNIFORM, RANDOM_NORMAL are random layouts where vertex positions are sampled from the uniform and the normal distributions, respectively.

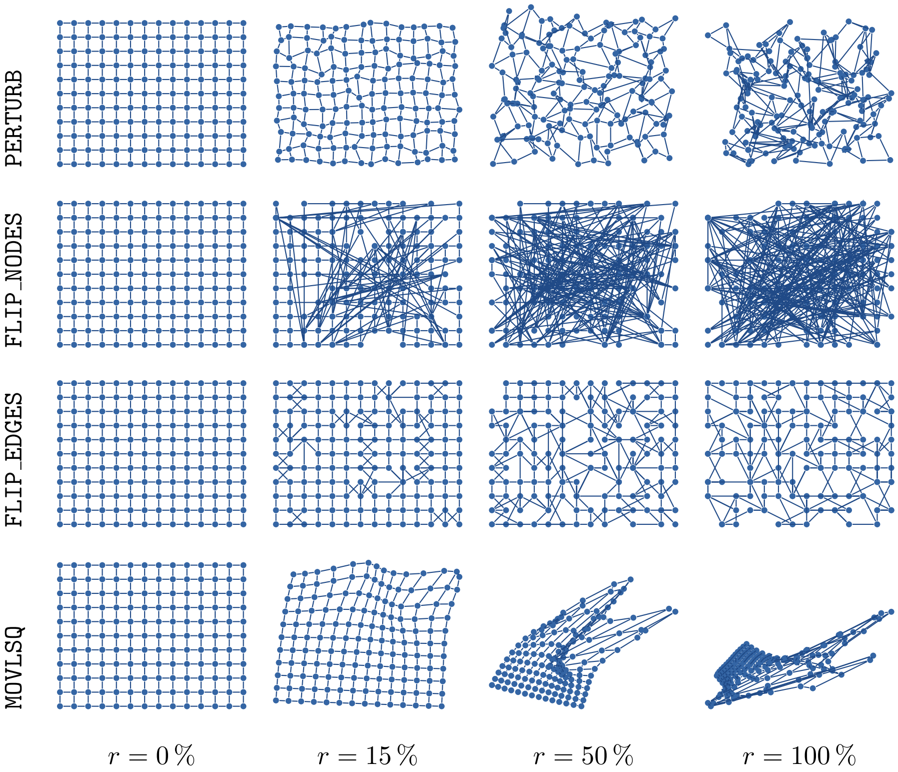


Figure 9: Examples of applying different layout worsening techniques at different rates.

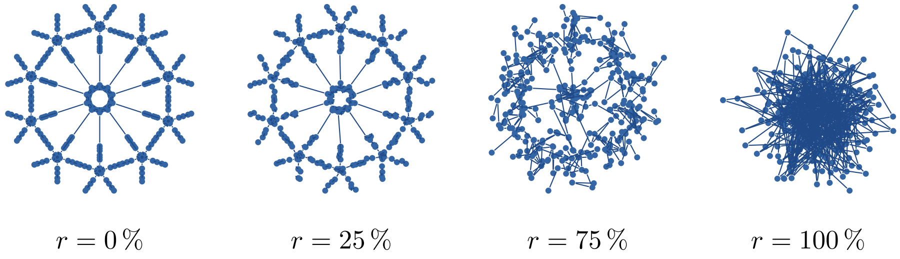


Figure 10: Example of linear interpolation between a proper and a garbage layout.

Layout Interpolation: As the second data augmentation technique, we linearly interpolated the positions of corresponding vertices between the proper and garbage layouts of the same graph. The resulting label t is then proportional to the difference in the interpolation parameter. Fig. 10 shows an example of linear interpolation between two layouts.

In total, using all the methods described above, we have been able to collect a database of about 76 000 labeled layout pairs.

8 Evaluation

The performance of the discriminator model was evaluated using *cross-validation* with 10-fold random subsampling [29]. In each round, 20 % of graphs (with all their layouts) were chosen randomly and were set aside for testing, and the model was trained using the remaining layout pairs. Of N labeled pairs used for testing, in each round we computed the number N_{correct} of pairs for which the model properly predicted the aesthetic preference, and derived the accuracy (success rate) $A = N_{\text{correct}}/N$. The standard deviation of A over the 10 runs was taken as the uncertainty of the results. With the average number of test samples of $N = 15554$, the eventual success rate was $\boldsymbol{A = (97.58 \pm 0.41)\,\%}$.

8.1 Comparison With Other Metrics

In order to assess the relative standing of the suggested method, we have implemented two known aesthetic metrics (*stress* and the *combined metric* by Huang et al. [23]) and evaluated them over the same dataset. The metric values were trivially converted to the respective discriminator function outputs.

Stress $\mathcal{T}$ of a layout Γ of a simple connected graph $G = (V, E)$ was defined by Kamada and Kawai [24] as

$$\mathcal{T}(\Gamma) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} k_{ij} \left(\operatorname{dist}_\Gamma(v_i, v_j) - L \operatorname{dist}_G(v_i, v_j)\right)^2 , \tag{6}$$

where L denotes the desirable edge length and $k_{ij} = K/\operatorname{dist}_G(v_i, v_j)^2$ is the strength of a "spring" attached to v_i and v_j. The constant K is irrelevant in the context of discriminator functions and can be set to any value.

As observed by Welch and Kobourov [54], the numeric value of stress depends on the layout scale via the constant L in the Eq. 6 which complicates comparisons. Their suggested solution was for each layout to find L that minimizes $\mathcal{T}$ (e.g. using binary search). In our implementation, we applied a similar technique based on fitting and minimizing a quadratic function to the stress computed at three scales. We refer to this quantity as `STRESS`.

The combined metric proposed by Huang et al. [23] (referred to as `COMB`) is a weighted average of four simpler quality metrics: the number of edge crossings (`CC`), the minimum crossing angle between any two edges in the drawing (`CR`), the minimum angle between two adjacent edges (`AR`), and the standard deviation computed over all edge lengths (`EL`).

The average is computed over the so-called z-scores of the above metrics. Each z-score is found by subtracting the mean and dividing by the standard deviation of the metric for all layouts of a given graph to be compared with each other. More formally, let G be a graph and $\Gamma_1, \ldots, \Gamma_k$ be its k layouts to be compared pairwise. Let $M(\Gamma_i)$ be the value of metric M for Γ_i and μ_M and σ_M be the mean and the standard deviation of $M(\Gamma_i)$ for $i \in \{1, \ldots, k\}$. Then

$$z_M^{(i)} = \frac{M(\Gamma_i) - \mu_M}{\sigma_M} \tag{7}$$

is the z-score for metric M and layout Γ_i. The combined metric then is

$$\texttt{COMB}(\Gamma_j) = \sum_M w_M \, z_M^{(j)} \,. \tag{8}$$

The weights w_M were found via Nelder-Mead maximization [34] of the prediction accuracy over the training dataset[§].

The accuracy of the stress-based and the combined model-based discriminators is shown in Tab. 1. In most cases, our model outperforms these algorithms by a comfortable margin of $2-3\%$. Fig. 11 provides examples of mis-predictions. By inspecting such cases, we notice that `STRESS` often fails to guess the aesthetics of (almost) planar layouts that contain both very short and very long edges (such behavior may also be inferred from the definition of `STRESS`). It is known that for some planar graphs, such as nested triangulations, this property is unavoidable in planar drawings. The mis-predictions of `COMB` seem to be due to the high weight of the edge length metric `EL`. Both `STRESS` and `COMB` are weaker than our model in capturing the absolute symmetry and regularity of layouts.

8.2 Significance of Individual Syndromes

In order to estimate the influence of individual syndromes on the final result, we have tested several modifications of our model. For each syndrome, we

[§] The obtained weights are: $w_{\texttt{CC}} = +0.6929 \pm 0.2521$, $w_{\texttt{EL}} = +0.2803 \pm 0.2263$, $w_{\texttt{CR}} = -0.0216 \pm 0.0210$, $w_{\texttt{AR}} = -0.0051 \pm 0.0049$.

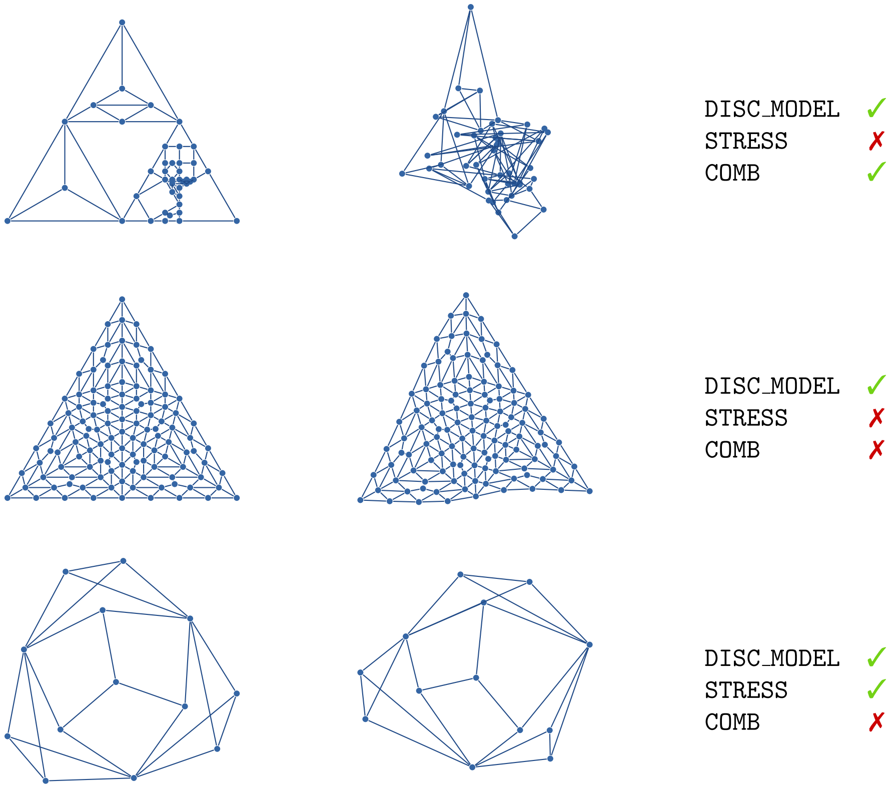


Figure 11: Examples where our discriminator model (DISC_MODEL) succeeds (✓) and the competing metrics fail (✗) to predict the answer correctly. In each row, the layout on the left is expected to be superior compared to the one on the right.

Metric	*Success Rate*	*Advantage*
COMB	(95.42 ± 1.15) %	(2.16 ± 1.11) %
STRESS	(94.97 ± 0.37) %	(2.61 ± 0.44) %

Table 1: Accuracy scores for the COMB and STRESS model. The standard deviation in each column is estimated based on the 5-fold cross-validation (using 20 % of data for testing each time). The "Advantage" column shows the improvement in the accuracy of our model with respect to the alternative metric.

considered the case when the feature vector contained only that syndrome. In the second case, that syndrome was removed from the original feature vector. The entries for the omitted features were set to zero. The results are shown in Tab. 2.

Property	*Sole Exclusion*	*Sole Inclusion*
PRINCOMP1	(97.29 ± 0.57) %	(54.75 ± 1.88) %
PRINCOMP2	(97.24 ± 0.32) %	(59.93 ± 3.63) %
ANGULAR	(97.11 ± 0.43) %	(73.28 ± 9.28) %
EDGE_LENGTH	(97.62 ± 0.35) %	(73.77 ± 6.90) %
RDF_GLOBAL	(97.47 ± 0.48) %	(83.43 ± 3.84) %
TENSION	(97.44 ± 0.53) %	(88.48 ± 1.61) %
RDF_LOCAL	(83.60 ± 4.67) %	(96.22 ± 1.29) %
Baseline Using All Properties	(97.58 ± 0.41) %	

Table 2: Success rates of our discriminator when a syndrome is excluded from the feature vector, and when the feature vector contains only that a syndrome. Note that RDF_LOCAL is a family of syndromes that are all included or excluded together. The apparent paradox of higher success rates when some syndromes are excluded can be explained by a statistical fluctuation; the respective "improvements" are well within the estimated uncertainty.

As can be observed, the dominant contribution to the accuracy of the model is due to the RDF-based properties RDF_LOCAL and RDF_GLOBAL. The exclusion of other syndromes does not significantly change the results (they agree within the estimated uncertainty). However, the sole inclusion of these syndromes still performs better than random choice (50 % success rate). This suggests that there is a considerable overlap between the aesthetic aspects captured by various syndromes. Further analysis is needed to identify the nature and the magnitude of these correlations. Superficially, Table 2 may imply that all the features except for RDF_LOCAL can be neglected. However, since RDF_LOCAL is the most computationally expensive feature, this would not significantly affect the runtime.

9 Conclusion

In this paper we propose a machine learning-based discriminator model that selects the more aesthetically pleasing drawing from a pair of graph layouts. Our model picks the "better" layout in more than 97 % cases and outperforms known stress-based and linear combination-based models. To the best of our knowledge, this is the first application of machine learning methods to this question. Previously, such techniques have proven successful in a range of complex issues involving aesthetics, prior knowledge, and unstated rules in object recognition, industrial design, and digital arts. As our model uses a simple network architecture, investigating the performance of more complex networks is warranted.

Previous efforts were focused on determining the aesthetic quality of a layout as a weighted average of individual quality metrics. We extend these ideas and findings in the sense that we do not assume any particular form of dependency between the overall aesthetic quality and the individual quality metrics.

Moving beyond simple quality metrics, we define quality syndromes that capture arrays of information about graphs and layouts. In particular, we borrow the notion of RDF from Statistical Physics and Crystallography; RDF-based features demonstrate the strongest potential in extracting the aesthetic quality of a layout. We expect RDFs (describing the microscopic structure of materials) to be the most relevant for large graphs. It is tempting to investigate whether further tools from physics can be useful in capturing drawing aesthetics.

From multiple syndromes, we construct fixed-size feature vectors using common statistical tools. Note that our feature vectors do not contain any explicit information about crossings or crossing angles. Nevertheless, the resulting performance is superior with respect to the weighted averages-based model which accounts for both. Further, incorporating symmetry measures recently suggested by Purchase [38] and Klapaukh [28] could be an interesting option.

In order to train and evaluate the model, we have assembled a relatively large corpus of labeled pairs of layouts, using available and generated graphs and exploiting the assumption that layouts produced by force-directed algorithms and native graph layouts are aesthetically pleasing and that random disturbances reduce their aesthetic quality.

Of course, to achieve a higher practical relevance, this study should ideally be repeated with human-labeled data. However, this requires that a dataset be collected with a size similar to ours, which is a challenging task. Building infrastructure to attract people, collect responses, and process the results, as well as data collection campaigns per se may require considerable resources and efforts. Nevertheless, we believe that creating such a dataset would be a critically important accomplishment in the graph drawing field.

Another open question is how to efficiently incorporate explicit and implicit domain-specific rules and expert knowledge into models (domain adaptation). A fully representative collection of graph drawings related to some domain by definition encodes all field-specific conventions and preferences. Therefore, a sufficiently flexible model trained on this dataset automatically adapts to the domain. Often, however, available datasets are not representative, have unknown biases, and contain too few samples illustrating some important rules. Different machine learning approaches admit various ways of correcting these biases and artificially increasing the importance of such “prior knowledge”. Studying these methods and developing convenient field-specific solutions would be an interesting extension of the present work.

References

[1] H. J. C. Barbosa and A. M. S. Barreto. An interactive genetic algorithm with co-evolution of weights for multiobjective problems. In L. Spector, E. D. Goodman, A. Wu, W. B. Langdon, and H.-M. Voigt, editors, *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, GECCO'01, pages 203–210. Morgan Kaufmann Publishers Inc., 2001.

[2] D. E. Berlyne. Novelty, complexity, and hedonic value. *Perception & Psychophysics*, 8(5):279–286, Sep 1970. `doi:10.3758/BF03212593`.

[3] R. F. Boisvert, R. Pozo, K. Remington, R. F. Barrett, and J. J. Dongarra. Matrix market: a web resource for test matrix collections. In R. F. Boisvert, editor, *Quality of Numerical Software: Assessment and enhancement*, pages 125–137. Springer, 1997. `doi:10.1007/978-1-5041-2940-4\_9`.

[4] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. *Advances in Neural Information Processing Systems*, pages 737–744, 1994. `doi:10.1142/S0218001493000339`.

[5] C.-C. Carbon, T. Mchedlidze, M. H. Raab, and H. Waechter. The power of shape: How shape of node-link diagrams impacts aesthetic appreciation and triggers interest. *i-Perception*, 9, 2018. `doi:10.1177/2041669518796851`.

[6] M. Chimani, C. Gutwenger, M. Jünger, G. W. Klau, K. Klein, and P. Mutzel. The open graph drawing framework (OGDF). In R. Tamassia, editor, *Handbook on Graph Drawing and Visualization.*, pages 543–569. Chapman and Hall/CRC, 2013.

[7] M. K. Coleman and D. S. Parker. Aesthetics-based graph layout for human consumption. *Softw. Pract. Exper.*, 26(12):1415–1438, Dec. 1996. `doi:10.1002/(SICI)1097-024X(199612)26:12<1415::AID-SPE69>3.0.CO;2-P`.

[8] K. Dev, N. Villar, and M. Lau. Polygons, points, or voxels?: stimuli selection for crowdsourcing aesthetics preferences of 3d shape pairs. In B. Gooch, Y. I. Gingold, H. Winnemoeller, L. Bartram, and S. N. Spencer, editors, *Proceedings of the symposium on Computational Aesthetics, CAE 2017, Los Angeles, California, USA*, pages 2:1–2:7. ACM, 2017. `doi:10.1145/3092912.3092918`.

[9] R. dos Santos Vieira, H. A. D. do Nascimento, and W. B. da Silva. The application of machine learning to problems in graph drawing – a literature review. In *Proceedings og the 7th International Conference on Information, Process, and Knowledge Management, eKNOW 2015, Lisbon, Portugal*, pages 112–118, 2015.

[10] P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 24:149–160, 1984.

[11] P. Eades, S. Hong, A. Nguyen, and K. Klein. Shape-based quality metrics for large graph visualization. *J. Graph Algorithms Appl.*, 21(1):29–53, 2017. `doi:10.7155/jgaa.00405`.

[12] G. H. Findenegg and T. Hellweg. *Statistische Thermodynamik*. Springer Spektrum, 2 edition, 2015. `doi:10.1007/978-3-642-37872-0`.

[13] B. L. Fredrickson. What good are positive emotions. *Review of General Psychology*, 2:300–319, 1998. `doi:10.1037/1089-2680.2.3.300`.

[14] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Softw., Pract. Exper.*, 21(11):1129–1164, 1991. `doi:10.1002/spe.4380211102`.

[15] S. Hachul and M. Jünger. Drawing large graphs with a potential-field-based multilevel algorithm. In J. Pach, editor, *Proceedings of the 13th International Symposium on Graph Drawing, Limerick, Ireland*, pages 285–295. Springer Berlin Heidelberg, 2005. `doi:10.1007/978-3-540-31843-9\_29`.

[16] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405:947–951, 6 2000. `doi:10.1038/35016072`.

[17] H. Haleem, Y. Wang, A. Puri, S. Wadhwa, and H. Qu. Evaluating the readability of force directed graph layouts: A deep learning approach. *CoRR*, abs/1808.00703, 2018. URL: `http://arxiv.org/abs/1808.00703`, `arXiv:1808.00703`.

[18] W. Huang and P. Eades. How people read graphs. In S. Hong, editor, *Asia-Pacific Symposium on Information Visualisation, APVIS 2005, Sydney, Australia*, pages 51–58, 2005.

[19] W. Huang, P. Eades, S.-H. Hong, and C.-C. Lin. Improving multiple aesthetics produces better graph drawings. *Journal of Visual Languages and Computing*, 24(4):262–272, 2013. `doi:10.1016/j.jvlc.2011.12.002`.

[20] W. Huang, S. Hong, and P. Eades. Effects of crossing angles. In *IEEE VGTC Pacific Visualization Symposium 2008, PacificVis 2008, Kyoto, Japan*, pages 41–46, 2008. `doi:10.1109/PACIFICVIS.2008.4475457`.

[21] W. Huang, S.-H. Hong, and P. Eades. Layout effects on sociogram perception. In P. Healy and N. S. Nikolov, editors, *Graph Drawing*, pages 262–273, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. `doi:10.1007/11618058_24`.

[22] W. Huang and M. Huang. Exploring the relative importance of crossing number and crossing angle. In G. Dai, K. Zhang, M. L. Huang, H. Wang, X. Yuan, L. Tao, and W. Chen, editors, *Proceedings of the 3rd International Symposium on Visual Information Communication*, VINCI '10, pages 1–8. ACM, 2010. `doi:10.1145/1865841.1865854`.

[23] W. Huang, M. L. Huang, and C.-C. Lin. Evaluating overall quality of graph visualizations based on aesthetics aggregation. *Information Sciences*, 330:444–454, 2016. SI:Visual Info Communication. doi:10.1016/j.ins.2015.05.028.

[24] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989. doi:10.1016/0020-0190(89)90102-6.

[25] Keras. URL: https://keras.io/.

[26] M. Klammler. *Aesthetic value of graph layouts: Investigation of statistical syndromes for automatic quantification.* Master's thesis, Karlsruhe Institute of Technology, 2018. URL: http://klammler.eu/msc/.

[27] M. Klammler et al. Source code for aesthetic discrimination of graph layouts. URL: https://github.com/5gon12eder/msc-graphstudy.

[28] R. Klapaukh. *An Empirical Evaluation of Force-Directed Graph Layout.* PhD thesis, Victoria University of Wellington, 2014.

[29] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI 95, Montréal Québec, Canada, 2 Volumes*, pages 1137–1145. Morgan Kaufmann, 1995.

[30] O. H. Kwon, T. Crnovrsanin, and K. L. Ma. What would a graph look like in this layout? A machine learning approach to large graph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):478–488, Jan 2018. doi:10.1109/TVCG.2017.2743858.

[31] T. Masui. Evolutionary learning of graph layout constraints from examples. In P. A. Szekely, editor, *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology*, UIST '94, pages 103–108. ACM, 1994. doi:10.1145/192426.192468.

[32] M. Nishiyama, T. Okabe, Y. Sato, and I. Sato. Sensation-based photo cropping. In W. Gao, Y. Rui, A. Hanjalic, C. Xu, E. G. Steinbach, A. El-Saddik, and M. X. Zhou, editors, *Proceedings of the 17th ACM International Conference on Multimedia*, MM '09, pages 669–672. ACM, 2009. doi:10.1145/1631272.1631384.

[33] D. A. Norman. Emotion & design: attractive things work better. *Interactions*, 9(4):36–42, 2002. doi:10.1145/543434.543435.

[34] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, 3 edition, 2007.

[35] P. Prusinkiewicz and A. Lindenmayer. *The Algorithmic Beauty of Plants*. Springer, 1990. doi:10.1007/978-1-4613-8476-2.

[36] H. C. Purchase. Which aesthetic has the greatest effect on human understanding? In G. Di Battista, editor, *Proceedings of the 5th International Symposium on Graph Drawing, GD 1997, Rome, Italy*, volume 1353 of *Lecture Notes in Computer Science*, pages 248–261, 1997. doi:10.1007/3-540-63938-1_67.

[37] H. C. Purchase. Performance of layout algorithms: Comprehension, not computation. *Journal of Visual Languages and Computing*, 9(6):647–657, 1998. doi:10.1006/jvlc.1998.0093.

[38] H. C. Purchase. Metrics for graph drawing aesthetics. *J. Vis. Lang. Comput.*, 13(5):501–516, 2002. doi:10.1006/jvlc.2002.0232.

[39] H. C. Purchase, J.-A. Allder, and D. A. Carrington. Graph layout aesthetics in UML diagrams: User preferences. *J. Graph Algorithms Appl.*, 6:255–279, 2002. doi:10.7155/jgaa.00054.

[40] H. C. Purchase, R. F. Cohen, and M. James. Validating graph drawing aesthetics. In F. Brandenburg, editor, *Proceedings of the 4th International Symposium on Graph Drawing, GD 1996, Passau, Germany*, volume 1027 of *Lecture Notes in Computer Science*, pages 435–446. Springer Berlin Heidelberg, 1996. doi:10.1007/BFb0021827.

[41] H. C. Purchase, J. Hamer, M. Nöllenburg, and S. G. Kobourov. On the usability of Lombardi graph drawings. In W. Didimo and M. Patrignani, editors, *Proceedings of the 20th International Symposium on Graph Drawing, GD 2012, Redmond, USA*, volume 7704, pages 451–462. Springer Berlin Heidelberg, 2012. doi:10.1007/978-3-642-36763-2_40.

[42] A. Rosete-Suarez, M. Sebag, and A. Ochoa-Rodriguez. A study of evolutionary graph drawing, 1999. Laboratoire de Recherche en Informatique (LRI), Universite Paris-Sud XI, Tech. Rep. 1228.

[43] S. Schaefer, T. McPhail, and J. Warren. Image deformation using moving least squares. *ACM Trans. Graph.*, 25(3):533–540, 2006. doi:10.1145/1141911.1141920.

[44] D. W. Scott. On optimal and data-based histograms. *Biometrika*, 66(3):605–610, 1979. doi:10.1093/biomet/66.3.605.

[45] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(4):623–656, 10 1948. doi:10.1002/j.1538-7305.1948.tb00917.x.

[46] M. Spönemann, B. Duderstadt, and R. von Hanxleden. Evolutionary meta layout of graphs. In T. Dwyer, H. C. Purchase, and A. Delaney, editors,

Proceedings of 8th International Conference on Diagrammatic Representation and Inference, Diagrams 2014, Melbourne, VIC, Australia, pages 16–30. Springer Berlin Heidelberg, 2014. doi:10.1007/978-3-662-44043-8_3.

[47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[48] R. Tamassia. *Handbook of Graph Drawing and Visualization.* Discrete Mathematics and Its Applications. CRC Press, 2013.

[49] Tensor flow. URL: https://tensorflow.org/.

[50] N. Tractinsky, A. S. Katz, and D. Ikar. What is beautiful is usable. *Interacting with Computers*, 13(2):127–145, 2000. doi:10.1016/S0953-5438(00)00031-X.

[51] M. S. Treder. Behind the looking-glass: A review on human symmetry perception. *Symmetry*, 2(3):1510–1543, 2010. doi:10.3390/sym2031510.

[52] C. Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2012.

[53] C. Ware, H. C. Purchase, L. Colpoys, and M. McGill. Cognitive measurements of graph aesthetics. *Information Visualization*, 1(2):103–110, 2002. doi:10.1057/palgrave.ivs.9500013.

[54] E. Welch and S. Kobourov. Measuring symmetry in drawings of graphs. *Computer Graphics Forum*, 36(3):341–351, 2017. doi:10.1111/cgf.13192.