Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

Data-enabled sketch search and retrieval for visual design stimuli generation

Zijian Zhang and Yan Jin

Department of Aerospace & Mechanical Engineering, University of Southern California, Los Angeles, CA 90089, USA

Research Article

Cite this article: Zhang Z, Jin Y (2022). Data-enabled sketch search and retrieval for visual design stimuli generation. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **36**, e25, 1–18. https://doi.org/10.1017/S0890060422000063

Received: 28 January 2021
Revised: 21 April 2022
Accepted: 19 June 2022

Key words:
Data-enabled design; design-by-analogy; sketching; unsupervised deep learning; visual fixation

Author for correspondence:
Yan Jin,
E-mail: yjin@usc.edu

Abstract

Access to vast datasets of visual and textual materials has become significantly easier. How to take advantage of the conveniently available data to support creative design activities remains a challenge. In the phase of idea generation, the visual analogy is considered an effective strategy to stimulate designers to create innovative ideas. Designers can read useful information off vague and incomplete conceptual visual representations, or stimuli, to reach potential visual analogies. In this paper, a computational framework is proposed to search and retrieve visual stimulation cues, which is expected to have the potential to help designers generate more creative ideas by avoiding visual fixation. The research problems include identifying and detecting visual similarities between visual representations from various categories and quantitatifying the visual similarity measures serving as a distance metric for visual stimuli search and retrieval. A deep neural network model is developed to learn a latent space that can discover visual relationships between multiple categories of sketches. In addition, a top cluster detection-based method is proposed to quantify visual similarity based on the overlapped magnitude in the latent space and then effectively rank categories. The QuickDraw sketch dataset is applied as a backend for evaluating the functionality of our proposed framework. Beyond visual stimuli retrieval, this research opens up new opportunities for utilizing extensively available visual data as creative materials to benefit design-by-analogy.

Introduction

Data-enabled design is recently proposed to use big data as design materials to probe for user behaviors and integrate them into design processes (Bogers *et al.*, 2016). The early-stage design process is regarded as an explorative activity in which visual thinking plays an important role (Goldschmidt, 1994). The visual analogy is a visual reasoning method that has been used by designers to generate creative ideas (Goldschmidt, 2001; Casakin, 2010). Many analogy search and retrieval computational tools have been developed to discover inspirations based on the idea that designers need large datasets containing design precedents to support the evocation process (Setchi and Bouchard, 2010; Chakrabarti *et al.*, 2017; Han *et al.*, 2018). While current search engines allow designers to utilize keywords to initiate searching based on semantic similarity, few computational tools exist for exploring sketches and images to find similar or drastically different visual stimulation cues to support visual analogy. How to identify and present valuable analogies from vast sets of visual materials to a designer remains a problem for the research community.

Sketches, CAD, photographs, and line drawings are central visual representations to promote analogical thinking (Goldschmidt, 2001; Gonçalves *et al.*, 2014). Designers can take advantage of visual imagery to manipulate shapes and generate meaningful and even creative concepts. Sketches are abstract and ambiguous drawings widely used by designers to materialize their mental imagery while discarding unnecessary details. The advantage of displaying self-generated sketches is that it provides valuable cues for visual analogy (Goldschmidt and Smolkov, 2006). Following this line of thinking, we propose a novel approach to help enhance designers' visual analogy making by providing meaningful visual cues to the designers in response to their visual and textual queries based on the vast imagery datasets available. The key idea behind this approach is: if rich networks of *shape-connections* and *semantic-connections* among the imagery items, for example, images and sketches, in a massive dataset can be established, then the meaningful recommendations can be retrieved from the dataset to assist the designers' analogy making process.

There can be different types of shape connections, including shape patterns, stroke numbers (for sketches), and levels of image complexity. The semantic connections can be function-based, product category-based, or any assigned meaning-based. The challenge for realizing the visual analogy support is to allow designers to perform *visual* or *shape-based* (instead of keyword-based) searches that return relevant but non-obvious visual cues (instead of words or sentences) for designers' analogy making. As the first step to address the challenge, we take sketches as imagery items and seek to develop a deep learning-based framework that maps

© The Author(s), 2022. Published by Cambridge University Press

sketches into a shape feature-based space (as *shape connections*) and bridges multiple sketch categories (as *semantic connections*).

The Quickdraw dataset is a collection of 50 million sketches across 345 categories, which is contributed by players of the game Quick, Draw! (Jongejan, 2016). It can serve as a massive dataset of potential creative materials for searching for inspiration to motivate idea generation. Recently, machine learning models have been proposed to pull out visual patterns in the Quickdraw dataset and provide new ways to navigate the data (Chen *et al.*, 2017). Taking a dimension reduction approach similar to the previous work, in this paper, we map the very high-dimensional *sketch data* to a significantly low-dimensional space, called a *latent space*. In the latent space, the shape features from the multiple sketch categories can be learned, and relevant visual patterns can be identified and presented to designers.

After distilling the shape features, visual similarities between low-dimensional sketch representations in the latent space need to be measured when searching for visual analogies. Based on the visual similarity measures, the visual analogies can be retrieved, which are often not obviously identifiable by designers even when they are from heterogeneous categories. Analogy distance is the way to measure the similarity between the source domain and the target domain. It has been shown that long-distance analogies result in more creative ideas than short-distance analogies (Christensen and Schunn, 2007; Jin and Benami, 2010; Fu *et al.*, 2013*b*). It has been indicated that long- and short-distance analogies are qualitatively and subjectively determined by designers (Herring *et al.*, 2009; Casakin, 2010; Kwon *et al.*, 2019). Few researchers pay attention to quantifying the distance of visual analogies and rank potential stimuli based on the distance. Hence, the research problem in this paper is *how to identify and quantitatively measure visual similarity between sketches in the latent space, which can support detecting their visual relationships of the underlying structures, despite the differences in superficial features*.

In this paper, our goal is to develop a visual stimuli generation framework by utilizing deep learning techniques and quantitative methods to learn insights from human-generated sketches stored in accessible big data sources. Firstly, we automatically collect raw sketch data from QuickDraw and then devise an algorithm to compress these high-dimensional data to a latent space, in which shape patterns of each sketch category can be learned and represented. After that, a top clustering detection (TCD)-based method is proposed to quantify visual similarity and find visual relationships between categories in the latent space. Based on the learned visual similarities, given a sketch as a query, visually similar or dissimilar sketches within or beyond the same category can be searched, retrieved, and effectively presented to designers, potentially helping their visual analogy process.

Related work

Data sources for design by analogy

External stimuli available in various databases can benefit designers, especially those who have limited domain knowledge. Data sources for design by analogy can vary across different types and forms. In terms of representation modalities, stimuli can be pictorial or verbal/textual. Many computational tools and methods have been proposed to search and retrieve textual stimuli to support design-by-analogy. Patent databases and biology corpora are two primary data sources for textual stimuli. IDEA-INSPIRE was presented by Chakrabarti *et al.* to automatically search analogies from a biological database to solve a given design problem (Chakrabarti *et al.*, 2005; Sarkar and Chakrabarti, 2008). Shu *et al.* took advantage of natural language analysis to correlate functional basis terms with biological keywords extracted from biological knowledge available in books and journals (Chiu and Shu, 2007; Cheong *et al.*, 2011; Shu, 2010). Goel *et al.* proposed DANE (Design by Analogy to Nature Engine) to search and retrieve the functioning of biological systems in the Structure-Behavior-Function library to address engineering problems (Vattam *et al.*, 2011; Goel *et al.*, 2012). Luo *et al.* developed a data-driven tool to search technology knowledge from a patent database to detect design opportunities and directions (Luo *et al.*, 2017). Fu *et al.* presented a computational method to discover the functional and surface similarity of patents, and then designers can automatically retrieve analogical stimuli from these patents (Fu *et al.*, 2013*a*).

As designers are considered visualizers who are skillful in making and manipulating visual representations, they prefer visual stimuli. The use of visual stimuli through analogy is seen as a powerful strategy for designers to generate creative concepts. Jin and Benami indicated that the shapes and structures of a design artifact might be more stimulating than the functions (Jin and Benami, 2010). Goldschmidt *et al.* showed that designers surrounded by visual stimuli could produce more ideas with a high level of creativity (Goldschmidt and Smolkov, 2006). Linsey *et al.* showed that designers often prefer visual representations to textual descriptions for idea generation, and photographs are growing in popularity due to easy retrieval from the Internet (Linsey *et al.*, 2011; Atilola *et al.*, 2016). Du and MacDonald indicated that designers should be presented experimental product information as visual features that can be more easily recognized and recalled than text (Du and MacDonald, 2015). Goucher-Lambert *et al.* expanded upon why visual analogies can supplement or replace other modes of analogical transfer in design on the cognitive side (Goucher-Lambert *et al.*, 2019; Goucher-Lambert *et al.*, 2020).

Sketch as one of the visual inspiration sources significantly contributes to provoking idea generation. The display of sketches can help designers discover potentially visual hints that could help define a specific space in which a search for a creative concept is productive (Shah *et al.*, 2001). McKoy *et al.* showed that novice designers could generate higher quality and more novel design concepts when presented with sketches (McKoy *et al.*, 2001). Goel indicated that a sketch could be transformed into another visually related sketch but distinct from its previous sketch (Goel, 1995). Yang *et al.* suggested that the designer should create a drawing to elicit information from the user that will be useful in driving the design forward (Yang, 2009; Macomber and Yang, 2011). The power of sketches to promote visual analogy has been recognized by many researchers. The sketch is an ideal source to serve as a visual stimulus. However, few computational tools have been proposed to search and retrieve visual analogies from sketches effectively.

Sketch-based image retrieval

Researchers in the human–computer interaction and graphics communities have developed many sketch-based interactive tools to enable intuitive and rich user experiences, such as sketch-based image retrieval (SBIR). The goal of SBIR is to allow users to draw visual content and then find the most relevant examples in

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

large image databases. As keyword-based queries can become cumbersome when representing complex visual appearances, Sketch-based queries can be a complement to depicting the shape of an object (Chen *et al.*, 2009; Eitz *et al.*, 2010; Hu *et al.*, 2010; Cao *et al.*, 2013; Yu *et al.*, 2016).

Recent neuroscience work suggests that shape ambiguity can cause the human brain to recognize and reinterpret different objects from the same sketch (Walther *et al.*, 2011). Shape ambiguity can be regarded as shape similarity in this case, as separate objects can be abstracted into a common representation. An effective SBIR system should accurately retrieve relevant images in the same category as the query sketch. However, in this paper, we argue that SBIR needs to go beyond category recognition in engineering design. If designers want to search for images from a particular category, then keyword-based image search already gives them an efficient way to find vast amounts of relevant images. Designers often explore ideas in fields different from their original domain area, for example, designers of concept cars consider the trends in boat design. This process is highly subjective and personal. Hence, how to retrieve images from multiple categories based on visual similarity becomes a critical problem in our research.

There have been significant advancements in deep neural network models to learn a common and general feature space for sketches. QuickDraw is a massive dataset of human-drawn sketches obtained from an online game called Quick, Draw!. Sketch-rnn is an unsupervised learning model based on the Variational AutoEncoder (VAE) framework for extracting shape patterns from the sketches in QuickDraw database and generating similar sketches conditioned on learned shape features (Kingma and Welling, 2013; Ha and Eck, 2017). As a modification of sketch-rnn, Sketch-pix2seq is proposed to present shape features of multi-category objects in a latent space (Chen *et al.*, 2017). Recently, the authors have introduced dc-sketch-pix2seq which can reveal shape features for multiple categories of sketches and cluster sketches simultaneously (Zhang and Jin, 2020). This model has better performance for discovering visual similarities between sketch categories. In this paper, our dc-sketch-pix2seq is applied to learn a latent space that can represent the shape features of sketches. In the latent space, the visual similarity between different sketch categories can be quantified for measuring the analogical distance.

In summary, a rich body of research on computational methods has yet to be integrated into the extensive work on visual analogy. In this paper, we aim to fill the gap by developing a computational framework to search and retrieve visual stimuli from a comprehensive sketch database to support visual analogy.

Visual stimuli search and retrieval framework

A visual stimuli search and retrieval framework is proposed. Figure 1 shows the entire process, which consists of two main flows. The grey box includes the functions of the visual stimuli search and retrieval tool. The data-enabled visual analogy discovery flow along the white hollow arrows represents the processes of converting the design materials from visual stimuli sources to visual analogy knowledge. The visual stimuli search and retrieval flow along the solid blue arrows represents the processes of searching and retrieving visual analogies based on the visual analogy database when a designer needs to explore more opportunities and inspirations to avoid design fixation. There are three stages in the framework; each stage is explained as follows.

In stage 1, the previous design materials such as sketches, CAD drawings, photographs, and line drawings in the open-source datasets are collected and converted into images. The primary purpose of the proposed computational framework is to learn visual similarities between these images and apply the learned visual knowledge to discover visual analogies by proving a query image. Before learning visual similarities, the shape feature representations of the input images need to be extracted. This process can be done by compressing high-dimensional images to a latent space with much smaller dimensions but can represent the feature information of the inputs. In the latent space, all images from the data sources are represented by their shape features. All input images have no shape labels. Clustering is applied to group images in the latent space based on their shape features to detect the visual similarity of the images. After stage 1, the coordinates of all images in the latent space are determined, and the group of each image can be assigned. The detail of this step is explained in the section "Deep clustering encoder for extracting shape representations".

In stage 2, given the coordinates of all the source images in the latent space, we can measure the distance between the two groups to decide the visual similarity magnitude. Long-distance means a lower level of visual similarity between the two groups. Besides, some special images can also be applied to measure the visual similarity between the two groups. For example, in Figure 1, we can see group 1 and group 2 share some visually similar images. These special images are classified under different conditions, which is illustrated in the section "Sketch classifications". If two groups share more special images, it means that two groups have larger overlaps. Therefore, the visual similarity between the two groups can also be decided by measuring their overlapping magnitude. A quantification method is proposed to calculate visual similarity based on overlap magnitude, explained in the section "Top clustering detection based visual similarity measurement".

In stage 3, a designer wants to find visual stimuli with similar shapes as the object to be designed. The visual search and retrieval tool can convert the designer's initial sketches to low dimension representations. These representations are queries to the visual analogy database to search for visually similar images and the corresponding category names. The clusters of all images in the visual analogy dataset are also displayed for the designer to visualize their visual relationship. Designers can take the retrieved images and category information as visual cues to promote their visual thinking and then connect them to relevant but unrevealed design knowledge in their minds. They can take advantage of the retrieved category names to search for more interesting images in their mind, Google Image, or other databases. The visual relationship graph can provide designers with a map to explore the visual similarity between different groups.

Methodology

Technical abstraction of visual stimuli search and retrieval framework

The main idea of our computational framework is to search and retrieve sketches generated by humans as creative source materials and stimulate designers to create potentially more innovative concepts driven by visual analogy. To realize this, we propose steps for building the computational framework, which is shown in Figure 2. It consists of three major steps, including: (1) pretrain

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

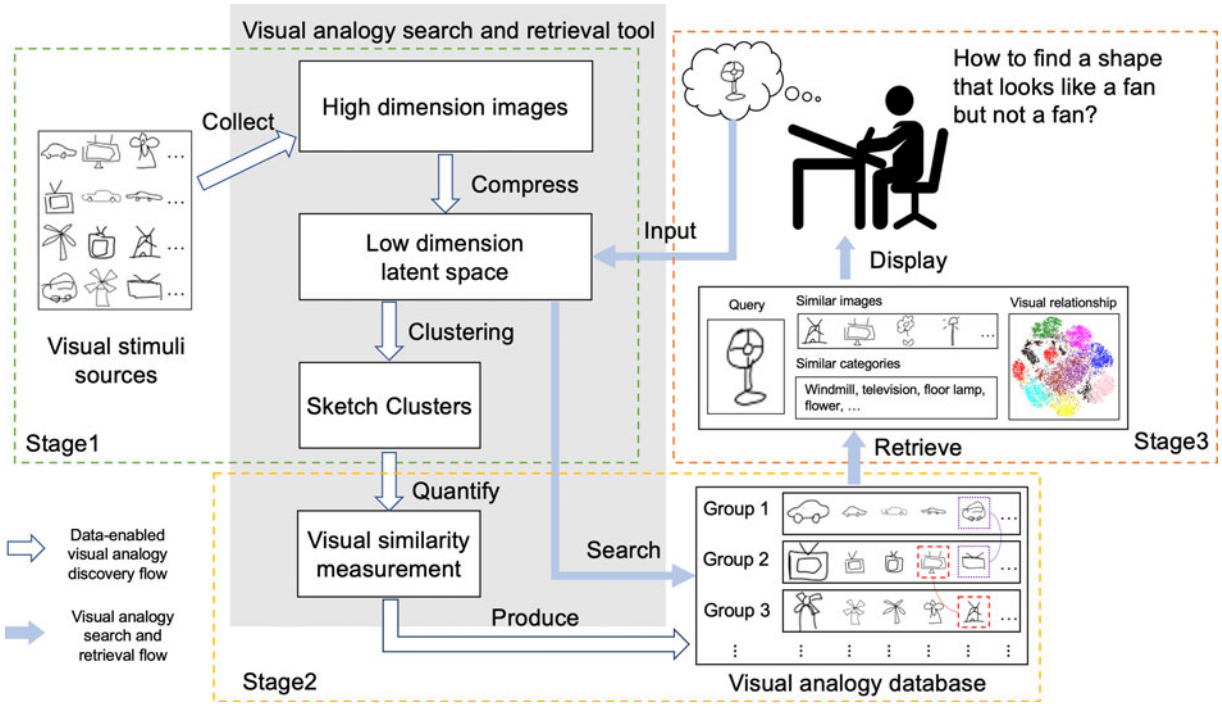


Fig. 1. Visual stimuli search and retrieval framework.

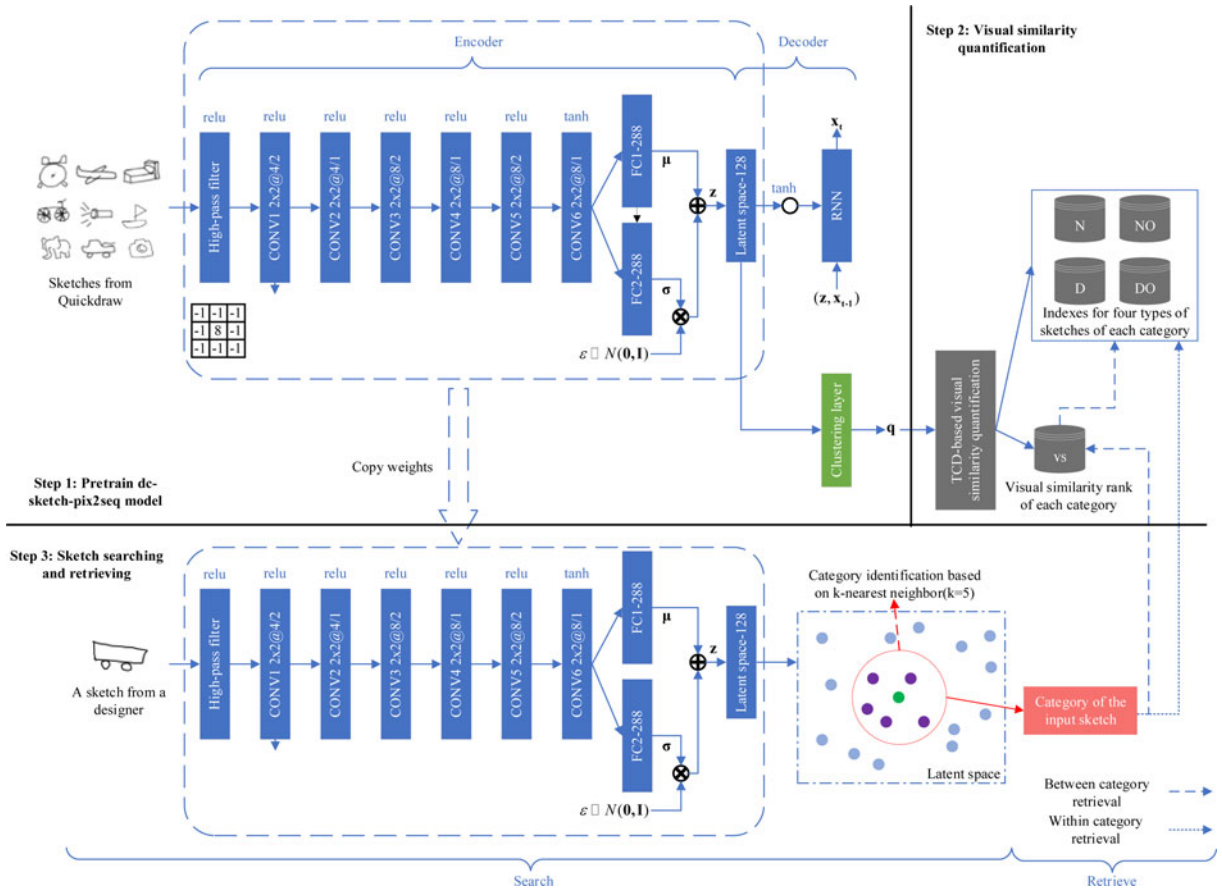


Fig. 2. Steps for building the visual stimuli search and retrieval framework.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

the dc-sketch-pix2seq model, (2) quantify visual similarities, and (3) search and retrieve relevant sketches.

In the first step, the training dataset of the dc-sketch-pix2seq model contains sketches with different categories from Quickdraw. The model can learn the weights of the encoder and decoder and a latent space in which the shape features of the training sketches can be captured and represented. After acquiring the low-dimensional representations (latent vectors) for the training sketches, the deep clustering layer can learn a clustering space in which each latent vector is clustered using a soft clustering method to generate probability distribution. The distribution can be used to determine which one or more clusters each latent vector belongs to.

In the second step, the TCD method is proposed to measure how many clusters one sketch may belong to quantitatively. Based on the results of TCD, sketches in each category can be classified into four different types: Native (N); Departed (D); Native-Overlap (NO); and Departed-Overlap (DO). The classified sketches can be saved in four databases with indexes. A N-sketch is useful to represent the shape of its category. A D-sketch, NO-sketch, or DO-sketch can be used to measure the visual similarity between the given category and other categories. Given the classification result in a given category, different weights are assigned to different types of sketches. By accumulating the weights, we can quantify the similarity magnitude and rank other categories in descending order. The rank can tell the most similar or least similar categories to the given category. Finally, calculate the visual similarity rank for each category and save the ranks in a database named *vs*.

In the third step, a designer draws a sketch that reflects the rough idea without details. The sketch is the input for the encoder of dc-sketch-pix2seq, which copies the weights from the learned encoder in step 1. The input can be mapped to the learned latent space to produce a latent vector that catches the shape features. In the latent space, the k-nearest neighbor algorithm is applied to find the nearest k latent vectors of the input sketch. The k latent vectors are from the training sketches, and their category labels are known. The input sketch will be labeled according to the most frequent label of these k latent vectors. Given the category label of the input sketch, we can retrieve the same category sketches from N, D, NO, and DO database; or retrieve *vs* database to find out categories with high and low similarity, and then retrieve sketches from N, D, NO, and DO database of other categories.

Deep clustering encoder for extracting shape representations

A shape representation extraction approach is taken to learn about the low-dimensional latent feature space of the given sketch datasets. More specifically, it is desired that a *generative model* can be trained that can discover embedded shape patterns of different sketch categories in the latent feature space without supervised information (e.g., category labels). Among various deep generative models, the variational autoencoder (VAE) is a widely used technique due to its good performance in learning a smooth latent representation of the input images.

Ha and Eck (2017) proposed a sequence-to-sequence VAE for generating sketch drawings for completing a user's stroke-based drawing sequence of common objects. In this model, the stroke-based sketch drawings are captured as a recurrent neural network (RNN) that can carry out conditional and unconditional sketch generation. Due to its stroke-based modeling approach, however, it has a key limitation, which is the insufficient quality of learning latent representations of sketches from *multiple* categories. The limitation made it inadequate to learn visual relationships between multiple categories.

To overcome the limitation of learning from single category sketches, Chen *et al.* (2017) replaced the RNN layers with CNN layers to deal with pixel-based sketches (i.e., images). This change also removed the limitation of single-category sketch drawings and made it possible for the CNN to learn from sketches of multiple categories and generate a wide variety of sketches based on the user's input. Since this paper considers visual analogies from multiple categories, it is important that our generative model learns from sketches of multiple categories. The deep learning-based sketch generative model, called the dc-sketch-pix2seq model, is defined as follows.

Figure 3 shows the model structure of dc-sketch-pix2seq. Before inputting into the convolutional neural network, a sketch is filtered by a 3×3 matrix on the bottom left to extract salient shape features. The configurations of convolutional layers are expressed as $h \times w@d/s$, where h, w, d, and s represent height, width, depth, and stride, respectively. The type of activate function is above each convolutional layer. The last convolutional layer outputs a one-dimensional vector, which is subsequently fed into two separate, fully connected layers. μ and σ are the mean and standard deviation of the posterior distribution, which is learned by the encoder, where $\boldsymbol{z} = \mu + \sigma \cdot \varepsilon$ is the latent vector and $\boldsymbol{x}$ is the input image. ε is a Gaussian noise. $\boldsymbol{z}$ is an input of the recurrent neural network defined in the paper (Ha and Eck, 2017), which is the decoder. The goal of the decoder is to reconstruct the output, which is almost identical to the input. A clustering layer is concatenated to the latent space to cluster sketches during the training of the model.

The input of dc-sketch-pix2seq is a set of n sketches $\boldsymbol{x} = \{x_i \in X\}_{i=1}^{n}$. X is the data space. The encoder compresses $\boldsymbol{x}$ into n latent vector $\boldsymbol{z} = \{z_i \in Z\}_{i=1}^{n}$. Z is the latent space. The dimensionality of Z is typically much smaller than X. The decoder samples n sketches $\boldsymbol{x}' = \{x'_i \in X\}_{i=1}^{n}$ conditional on given latent vector $\boldsymbol{z}$. The reconstruction loss of the dc-sketch-pix2seq model is:

$$L_{\mathrm{r}} = E_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}'|\boldsymbol{z})], \tag{1}$$

where $q(\cdot)$ denotes the encoder and $p(\cdot)$ denotes the decoder. ϕ and θ are the parameters to be trained in the encoder and decoder, respectively. The parameters are typically the weights and biases of the neural networks. $E_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}(\cdot)$ ensures the similarity between the generated strokes and the strokes within the sketches in the training set.

The clustering layer clusters all latent vectors in the latent space Z by simultaneously learning a set of K cluster centers $\{\mu_j \in Z\}_{j=1}^{K}$ and mapping each latent vector z_i into a soft label q_i by student's t-distribution (Maaten and Hinton, 2008). $\boldsymbol{q_i} = [q_{i1}, \ldots, q_{ij}, \ldots q_{ik}]$ is a soft label which quantifies the similarity between z_i and cluster center μ_j.

$$q_{ij} = \frac{(1 + z_i - \mu_j^2)^{-1}}{\sum_j (1 + z_i - \mu_j^2)^{-1}}, \tag{2}$$

where q_{ij} is the jth entry of $\boldsymbol{q_i}$, representing the probability of z_i belonging to cluster j.

The clustering loss L_{c} is defined as a KL divergence between the distribution of soft labels Q measured by student's

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

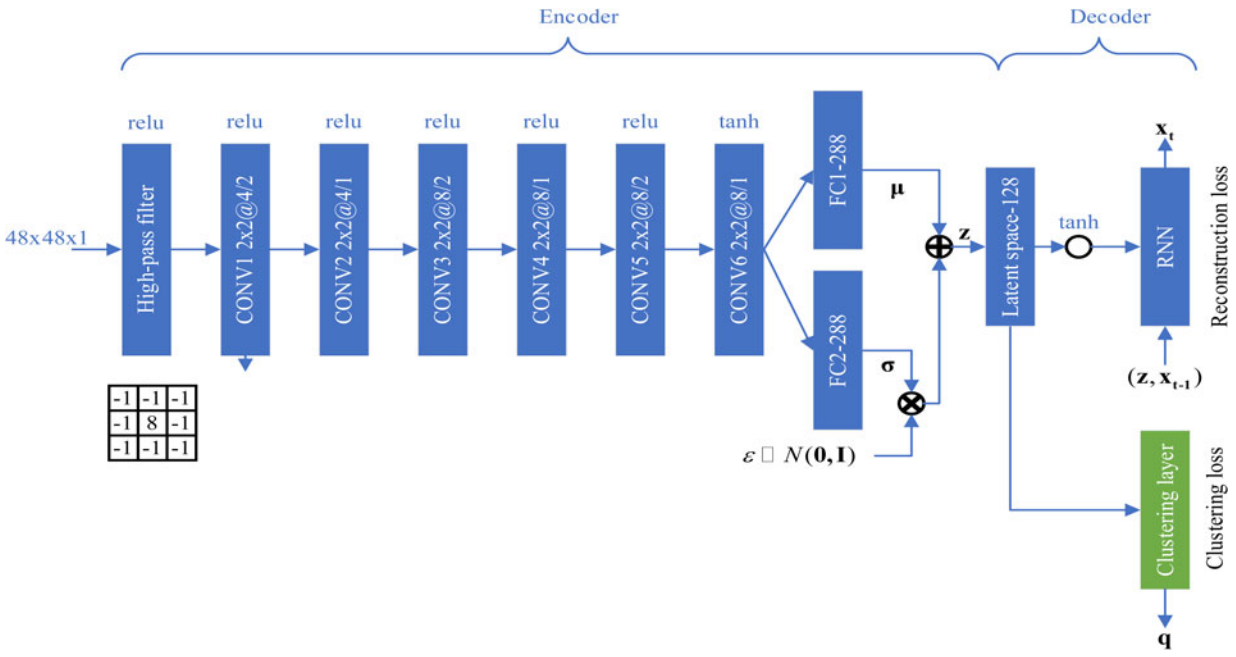


Fig. 3. Structure of deep clustering sketch-pix2seq.

t-distribution and the predefined target distribution P derived from Q. The clustering loss is defined as

$$L_c = D_{KL}(PQ) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}, \tag{3}$$

where the target distribution P is defined as

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_j \left(q_{ij}^2 / \sum_i q_{ij} \right)}. \tag{4}$$

Raising q_{ij} to the second power and then dividing by the frequency per cluster allows the target distribution P to improve cluster purity and stress on confident labels while normalizing the contribution of each centroid on the clustering loss to prevent large clusters from distorting the latent space. Therefore, data points with high confidence are used as supervision, and points in each cluster distribute more densely, where D_{KL} measures the nonsymmetric difference between two probability distributions, P, and Q are defined by (4) and (2), and match Q to P.

In dc-sketch-pix2seq, the encoder and decoder are used to learn representations in an unsupervised manner and the learned latent space can preserve shape features in sketches. The clustering loss is responsible for manipulating the latent space in order to cluster sketches based on visual similarity. The objective of dc-sketch-pix2seq is

$$L_{rc} = L_r + \tau L_c, \tag{5}$$

where L_r is a reconstruction loss from (2) and L_c is a clustering loss from (3). The coefficient τ is better to be less than 1 and more than 0.

The training process of dc-sketch-pix2seq and the performance comparison of dc-sketch-pix2seq with other two deep learning models (Chen *et al.*, 2017; Ha and Eck, 2017) are illustrated in our previous work (Zhang and Jin, 2020).

Sketch classifications

The output of the clustering layer is the probability distribution of each latent vector z_i into each soft clustering label j. By clustering space, we mean any K-dimensional vector $\boldsymbol{\rho} \in \mathbb{R}^K$ that represents a probability distribution of clustering, $\boldsymbol{\rho} = [p(c_1|\rho), \ldots, p(c_k|\rho), \ldots, p(c_K|\rho)]$, $c_k (1 \le k \le K)$ represents the jth cluster. $p(c_k|\rho)$ means the probability of data ρ belonging to kth cluster.

In dc-sketch-pix2seq, the inputs are sketches belonging to different categories, $\boldsymbol{x} = [x_{11}, \ldots, x_{ij}, \ldots, x_{st}]$, x_{ij} means the jth sketch belonging to ith category, s is the number of categories, t is the number of sketches in sth category. In the latent space, latent vectors are $\boldsymbol{z} = [z_{11}, \ldots, z_{ij}, \ldots, z_{st}]$. In the clustering space, the probability distributions of latent vectors can be represented by a super matrix $\mathbb{Q}$, $\mathbb{Q} = [\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_s]$. For matrix $\boldsymbol{Q}_i$ $(1 \le i \le s)$, it includes n sketches. $\boldsymbol{Q}_i = [\boldsymbol{q}_{i1}, \ldots, \boldsymbol{q}_{ij}, \ldots, \boldsymbol{q}_{in}]$, $\boldsymbol{q}_{ij} (1 \le j \le n)$ represents the probability distribution of a latent vector z_{ij}, that is $\boldsymbol{q_{ij}} = [p(c_1|z_{ij}), \ldots, p(c_k|z_{ij}), \ldots, p(c_K|z_{ij})]$, where $P(c_k|z_{ij})$ means the probability of z_{ij} belonging to cluster c_k, $\sum_1^K P(c_k|z_{ij}) = 1$. Soft clustering produces multi-cluster predictions for x_{ij}, while the ground truth category of x_{ij} is single labeled. For example, in a three-dimensional clustering space, assuming that all sketches also come from three categories, ω_1, ω_2, ω_3. When the ground truth category of x_{ij} is ω_i, the probability distribution of the corresponding latent vector z_{ij} might be $\boldsymbol{q_{ij}} = [0.8, 0.1, 0.1]$. The cluster prediction of z_{ij} is c_1 which has the maximum probability. However, if $\boldsymbol{q_{ij}} = [0.46, 0.45, 0.09]$, could we still say z_{ij} should be clustered to c_1 rather than both c_1 and c_2? Therefore, sketch x_{ij} might belong to more than one cluster. One sketch can belong to multiple clusters because it includes some shape features shared by various categories. It means these special sketches in the latent space can be helpful to calculate the visual similarity between different categories.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

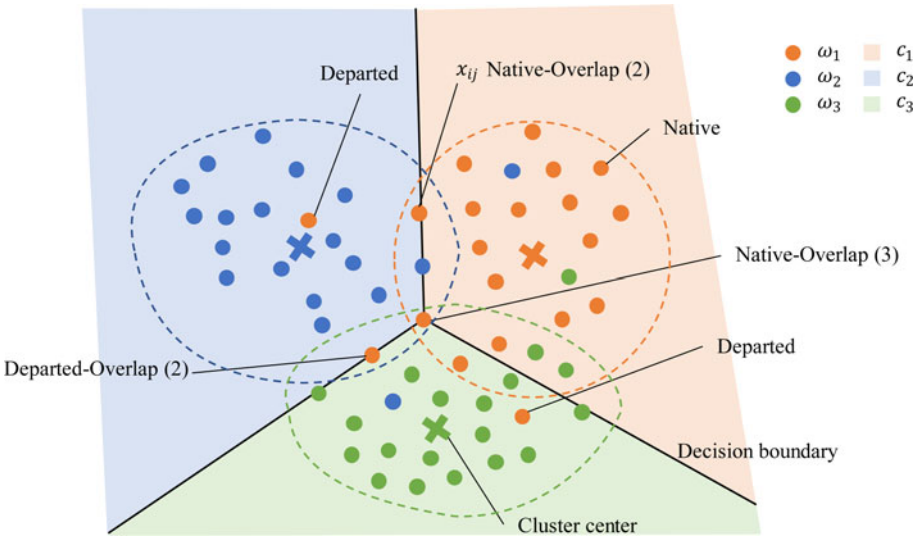


Fig. 4. Different types of sketches in a latent space including three clusters.

In Figure 4, circles "o" indicate input sketches and crosses "×" represent cluster centroids in the latent space. Different categories are rendered with different colors. Solid lines indicate decision boundaries that are perpendicular bisectors of adjacent cluster centers. There are some sketches from the categories ω_1, ω_2, ω_3. The cluster label of each category is determined by which cluster prediction most sketches belong. We call it the ground truth cluster. Supposed most sketches from the category ω_1 are predicted to belong to cluster c_1. Then, the ground truth cluster for category ω_1 is c_1. We assume sketches in the same category should have similar shapes and can be distinguished from other categories. Therefore, most sketches from the same category can be grouped in the same cluster based on shape similarity. The ground truth cluster is not assigned *a priori* but based on the location of most sketches from a category in the latent space. Each input sketch x_{ij} can be classified into four possible types, including:

- Native – only one clustering prediction that is the same as the ground truth cluster.
- Departed – only one clustering prediction that is not the same as the ground truth cluster.
- Native-Overlap (n) – multi-clustering predications that include the ground truth cluster; the overlap number n should be no more than the number of clusters.
- Departed-Overlap (n) – multi-clustering predictions that do not include the ground truth cluster; the overlap number n should be no more than the number of clusters.

In Figure 4, for category ω_1, all orange points belonging to cluster c_1 are native points; the orange points belonging to cluster c_2 or c_3 are departed points; the orange point located in the line between cluster c_1 and c_2 areas is a native-overlap point with two overlaps; the orange point located in the line between cluster c_1, c_2, and c_3 areas is a native-overlap point with three overlaps; the orange point located in the line between cluster c_2 and c_3 areas is a departed-overlap point with two overlaps.

Based on shape similarity, Native (N) points are representatives of a category. Native-Overlap (NO) points share similar shape features with other categories. Departed (D) points are "accidentally" clustered into another single category. It means the shapes of these sketches are more similar to another category. Departed-Overlap (DO) points are "accidentally" clustered into multiple other categories. It means the shapes of these sketches are more similar to other multiple categories. Possible reasons for why D and DO points appear as users of QuickDraw application are given a keyword to draw a sketch; they draw salient shape features of an object to represent the category. Some participants' visual understanding of the keyword may be different from most participants,' or they did not draw the sketch in the same shape pattern as most participants did.

Top clustering detection-based visual similarity measurement

In Figure 4, we can see points from the same category can be distributed in different clusters. It is not appropriate to use distances between cluster centroids to measure the similarity between different categories. For example, the centroid (orange) of the category ω_1 has almost the same distance to the centroid (blue) of the category ω_2 and the centroid (green) of category ω_3. Therefore, category ω_2 and category ω_3 have the same similarity with category ω_1 based on Euclidean distance. However, we can tell category ω_1 is more similar to category ω_3 than category ω_2 based on the overlap magnitude, as more sharing points exist in the overlap regions between category ω_1 and category ω_2. In this paper, we propose a novel method called top clustering detection (TCD) which can determine the most suitable number of top clustering for a sketch by finding the minimum Euclidean distance between the sketch with K ideal centers. K is the number of clusters. Generally, the ideal centers from 1-clustering to K-clustering are defined as follows.

$$\begin{bmatrix} l_1 \\ l_2 \\ \vdots \\ l_K \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}_{KK}. \quad (6)$$

Based on the TCD method, each sketch will be assigned to one or more than one clusters. Based on the assumption that sketches coming from the same category share more shape features. There is a high possibility that sketches in the same category will be assigned to the same cluster. The ground truth cluster of a

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

category is determined by which cluster label has the largest number of the top 1-clustering sketches. According to the previous section, we can classify sketches of each category into four types of points. Based on the type of points, we define the corresponding weights. According to the weights, we can calculate the similarity magnitude between different categories. The procedure to classify sketches into the category ω_i and calculate the visual similarity between category ω_i with other categories is shown as follows:

Step 1: In a K-dimensional clustering space, the probability distribution of a latent vector $\boldsymbol{z_{ij}}$ is

$$\boldsymbol{q_{ij}} = [p(c_1|\boldsymbol{z_{ij}}), \ldots, p(c_k|\boldsymbol{z_{ij}}), \ldots, p(c_K|\boldsymbol{z_{ij}})]. \tag{7}$$

Step 2: Find the maximum probability in $\boldsymbol{q_{ij}}$, then normalize $\boldsymbol{q_{ij}}$.

$$\bar{\boldsymbol{q}}_{ij} = \frac{\boldsymbol{q_{ij}}}{\max_k \boldsymbol{q_{ij}}}. \tag{8}$$

Step 3: Sort $\bar{\boldsymbol{q}}_{ij}$ in descending order to obtain $\bar{\boldsymbol{q}}^s_{ij}$.

Step 4: Argsort $\bar{\boldsymbol{q}}_{ij}$ in descending order to obtain $\bar{\boldsymbol{q}}'_{ij} = [c'_1, c'_2, \ldots, c'_K]$, where the topmost labels are more likely to be the multiple clustering labels of the input data x_{ij}.

Step 5: Calculate the Euclidean distance between $\bar{\boldsymbol{q}}^s_{ij}$ and l_k ($k = 1, 2, \ldots, K$), and the most possible number of clustering k'_{ij} is given by:

$$k'_{ij} = \operatorname*{argmin}_k \bar{\boldsymbol{q}}^s_{ij} - l_k, \tag{9}$$

where l_k is the ideal cluster center of the topmost k clustering, which is defined in (6).

Step 6: The clustering labels $\boldsymbol{c}_{ij}$ of x_{ij} is the first k' elements of $\bar{\boldsymbol{q}}'_{ij}$, namely $\boldsymbol{c}_{ij} = [c'_1, c'_2, \ldots, c'_{k'}]$. If $k' = 1$, x_{ij} is a top 1-cluster point. If $k' > 1$, x_{ij} is a multiple cluster point.

Step 7: Determine the ground truth cluster label of the category ω_i ($i = 1, 2, \ldots, s$) based on the most assigned cluster label of top 1-cluster points, assume it is c_i.

Step 8: Classify all input points into four types and determine weights. From step 6, assume all sketches in category ω_i are assigned by cluster label c_i. if x_{ij} is a native point $\boldsymbol{c}_{ij} = [c'_k]$ ($c'_k = c_i$), then add x_{ij} to a native point set N_i; if x_{ij} is a departed point $\boldsymbol{c}_{ij} = [c'_k]$ ($c'_k \neq c_i$), define the weight $w^{ij}_{c'_k}$ as 1, then add x_{ij} to a departed point set D_i; If x_{ij} is a native-overlap (k') point $\boldsymbol{c}_{ij} = [\underbrace{c'_1, c'_2, \ldots, c'_{k'}}_{k'}]$ (c_i is included), define the weight $w^{ij}_{c'_k}(c'_k \neq c_i)$ as $1/n$, then add x_{ij} to a native-overlap point set NO_i; If x_{ij} is a departed-overlap (k') point $\boldsymbol{c}_{ij} = [\underbrace{c'_1, c'_2, \ldots, c'_{k'}}_{k'}]$ (c_i is not included), define the weight $w^{ij}_{c'_k}$ as $1/n$, then add x_{ij} to a departed-overlap point set DO_i.

Step 9: For all sketches in the category ω_i, accumulate all weight values of other clusters $[\underbrace{c'_1, c'_2, \ldots, c'_K}_{K-1}]$ (c_i is not included). The cluster with the largest weight summation has the highest underlying shape similarity with cluster c_i, whose corresponding category is ω_i.

Step 10: Determine visual similarity. The similarity is computed by normalizing the weight summation by the largest number among all weights.

Step 11: Rank other clusters based on the weights in descending order to obtain visual similarity rank vs_i.

The TCD-based visual similarity quantification algorithm is shown as follows.

Algorithm 1 TCD-based visual similarity quantification

Input: Number of categories s; Number of sketches in each category t; Soft cluster assignments $\{\boldsymbol{q_{ij}} = [p(c_1|\boldsymbol{z_{ij}}), \ldots, p(c_k|\boldsymbol{z_{ij}}), \ldots, p(c_K|\boldsymbol{z_{ij}})]\}^{s,t}_{i=1,j=1}$

Output: Sketch classification $\{N_i\}^s_{i=1}$, $\{D_i\}^s_{i=1}$, $\{NO_i\}^s_{i=1}$, and $\{DO_i\}^s_{i=1}$; Visual similarity rank $\{vs_i\}^s_{i=1}$

for $i \in \{1, 2, \ldots, s\}$ **do**
  for $j \in \{1, 2, \ldots, t\}$ **do**
    normalize $\boldsymbol{q_{ij}}$ to obtain $\bar{\boldsymbol{q}}_{ij}$ using (8)
    sort $\bar{\boldsymbol{q}}_{ij}$ in a descending order to obtain $\bar{\boldsymbol{q}}^s_{ij}$
    argsort $\bar{\boldsymbol{q}}_{ij}$ in a descending order to obtain $\bar{\boldsymbol{q}}'_{ij}$
    find most possible number of clustering k'_{ij} of $\bar{\boldsymbol{q}}^s_{ij}$ using (9)
    determine soft clustering labels $\boldsymbol{c}_{ij}$ of x_{ij}
  end for
  determine the cluster label c_i of category ω_i
  classify x_{ij} into N_i, D_i, NO_i, or DO_i and determine weights $w^{ij}_{c'_k}$
  for $c'_k \in \{c'_1, c'_2, \ldots, c'_K\}$ **do**
    if $c'_k \neq c_i$ **then**
      $w^{\text{sum}}_{c'_k} = \sum_j w^{ij}_{c'_k}$
    end if
  end for
  nomalize and rank $w^{\text{sum}}_{c'_k}$ in descending order to obtain vs_i
end for

For example, in a latent space including three clusters, firstly, assuming the latent vector $\boldsymbol{z_{ij}}$ of the sketch x_{ij} from category ω_1 has a probability distribution $\boldsymbol{q_{ij}} = [0.5, 0.4, 0.1]$. After normalizing $\boldsymbol{q_{ij}}$ by the maximum probability of 0.5, we can have $\bar{\boldsymbol{q}}_{ij} = [1, 0.8, 0.2]$. Secondly, $\bar{\boldsymbol{q}}_{ij}$ is sorted in a descending order to generate $\bar{\boldsymbol{q}}^s_{ij} = [1, 0.8, 0.2]$. The cluster rank is $\bar{\boldsymbol{q}}'_{ij} = [1, 2, 3]$. Thirdly, the ideal centers for top 1-clustering, 2-clustering, and 3-clustering are $l_1 = [1, 0, 0]$, $l_2 = [1, 1, 0]$, and $l_3 = [1, 1, 1]$. The sketch x_{ij} will be assigned to the top k-clustering if l_k ($k = 1, 2$, or 3) is the nearest ideal center for the sketch. After calculation, the Euclidean distance between $\bar{\boldsymbol{q}}^s_{ij}$ with l_1, l_2, and l_3 are 0.82, 0.28, and 0.82, respectively. The possible number of clustering k'_{ij} is 2. It means the sketch x_{ij} will be assigned to the top 2-clustering of $\bar{\boldsymbol{q}}'_{ij}$. The clustering labels $\boldsymbol{c}_{ij} = [1, 2]$. x_{ij} is a multiple cluster point as it belongs to two clusters. Assume most top 1-clustering sketches in the category ω_1 are assigned to cluster c_1 and one category can be assigned to a unique cluster. Therefore, the ground truth cluster label of the category ω_i is 1. As $\boldsymbol{c}_{ij}$ includes cluster label 1, the type of x_{ij} is native-overlap (2). x_{ij} could be a point between category ω_1 and ω_2 in Figure 4. x_{ij} can be added to native-overlap point set NO_i and the weight $w^{ij}_{c_2} = 0.5$, which means the simialrity between cluster c_1 and cluster c_2 will be accumulated by 0.5 because of x_{ij}. For all skecthes in category ω_1, we can accumulate all weight values and find the similarity between category ω_1 with other categories. Take another

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

Table 1. Example of weights for four types points of category ω_1 in a latent space including three clusters

Data No.	Category	Ground truth cluster	Top k-clustering	Type	Weight
1	ω_1	c_1	c_1	Native	$w_{c_1}^{11} = 1$
2	ω_1	c_1	c_2	Departed	$w_{c_2}^{12} = 1$
3	ω_1	c_1	c_3	Departed	$w_{c_3}^{13} = 1$
4	ω_1	c_1	c_2, c_3	Departed-Overlap (2)	$w_{c_2}^{14} = \frac{1}{2}$, $w_{c_3}^{14} = \frac{1}{2}$
5	ω_1	c_1	c_1, c_3	Native-Overlap (2)	$w_{c_1}^{15} = \frac{1}{2}$, $w_{c_3}^{15} = \frac{1}{2}$
6	ω_1	c_1	c_1, c_2	Native-Overlap (2)	$w_{c_1}^{16} = \frac{1}{2}$, $w_{c_2}^{16} = \frac{1}{2}$
7	ω_1	c_1	c_1, c_2, c_3	Native-Overlap (3)	$w_{c_1}^{17} = \frac{1}{3}$, $w_{c_2}^{17} = \frac{1}{3}$, $w_{c_3}^{17} = \frac{1}{3}$

example, there are only seven points in the category ω_1. In a latent space including three clusters, the weights for these seven classified sketches in category ω_1 are shown in Table 1.

From the weight column, we can sum the weights of clusters c_2 and c_3, which are $w_{c_2}^{\text{sum}} = 7/3$, $w_{c_3}^{\text{sum}} = 7/3$. The higher weight summation means higher shape similarity with cluster c_1 which represents category ω_1. In this case, clusters c_2 and c_3 have same similarity with cluster c_1.

Experimental evaluation of sketch retrieval

In order to study the performance of visual stimuli search and retrieval based on the visual similarity of the proposed computational methods, we conduct several experiments on one dataset, which includes 10 categories from the Quickdraw dataset. The goals are to evaluate the following properties of the method for visual analogy support: (1) visualization effectiveness – how effective the potential stimuli can be visually presented to designers, (2) similarity measure quality – how the top clustering detection method yields better similarity measures comparing with traditional Euclidean distance methods, and (3) retrieval performance – the quality and efficiency of the sketch retrieval process.

Dataset and settings

The sketch data categories used for evaluation are television, canoe, drill, umbrella, car, floor lamp, guitar, windmill, wine bottle, and flower. The 75 K sketches for each category are divided into training, validation, and testing sets with 70, 2.5, and 2.5 K, respectively. The raw sketches from the Quickdraw dataset are converted to monochrome png files of size 48×48, which are used as the input data for the deep neural network model. The batch size is 100 for all datasets. The maximum number of epochs is set to 50. In each iteration, we train the encoder for one epoch using an Adam optimizer with a learning rate $\lambda = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. We implement the model end-to-end based on Python and Keras. The dimension of the latent space in the models is 128, which is the same as those used in Chen *et al.* (2017) and Ha and Eck (2017). For training the dc-sketch-pix2seq model, the coefficient τ of clustering loss in (5) is chosen as 0.05, which is determined by a grid search in {0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0} to evaluate different τ settings with unsupervised clustering accuracy (ACC). The ACC is defined as the best match between ground truth $\boldsymbol{y}$ and predicted cluster labels $\boldsymbol{c}$:

$$\text{ACC}(\boldsymbol{y}, \boldsymbol{c}) = \max_{m \in \mathcal{M}} \frac{\sum_{i=1}^{n} 1\{y_i = m(c_i)\}}{n}, \tag{10}$$

where n is the total number of samples, y_i is the ground truth label, c_i is the predicted cluster label of the example x_i obtained by the model, and $\mathcal{M}$ is the set of all possible one-to-one mappings between predicted cluster labels to ground truth cluster. The best cluster assignment can be efficiently computed by the Hungarian algorithm (Kuhn, 1955). In Figure 5, when $\tau = 0.05$, the dc-sketch-pix2seq model has the best clustering performance under the given value settings.

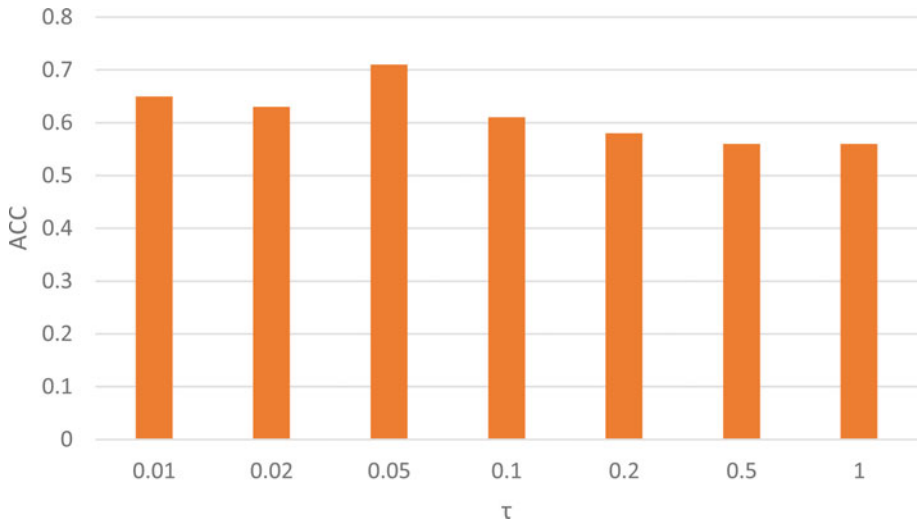


Fig. 5. ACC values under different τ value settings.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

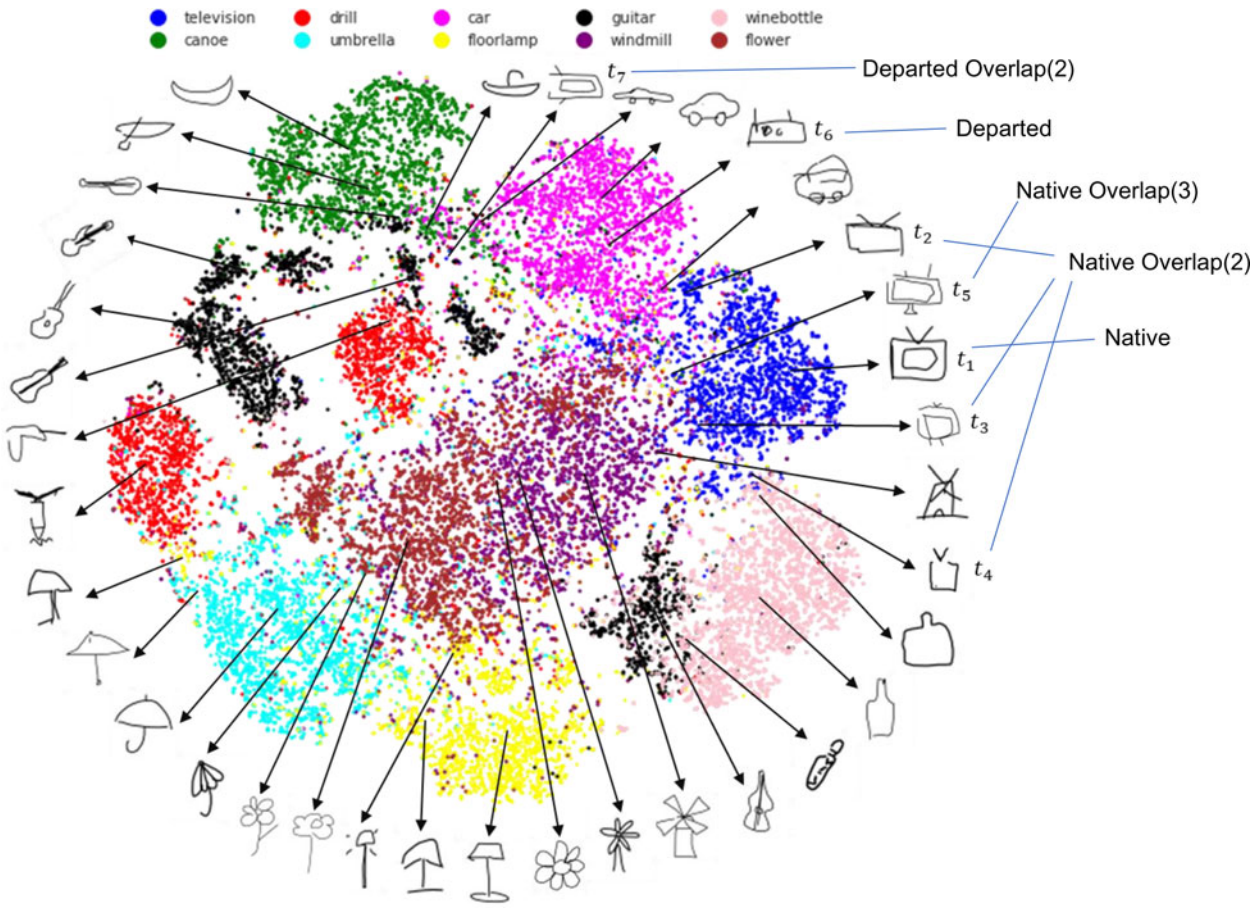


Fig. 6. Visualization of the latent space of 10 categories and different types of sketches.

A visualization of the learned features

The dimensionality reduction technique t-SNE (t-Distributed Stochastic Neighbor Embedding) (Maaten and Hinton, 2008) is applied to present a visualization of the learned latent space of 10 categories by projecting 128 dimensions to 2 dimensions. As shown in Figure 6, different color dots denote the sketches from different categories. Sketches in the same category are clustered in close proximity. The illustration in Figure 6 affirms our assumption that sketches in the same group have similar shape features and they are more likely clustered in the same group.

The overlap regions between different categories make it possible to find out the sharing shape features. In Figure 7, each category has 2,500 sketches. Most sketches of each category are native sketches with 0 overlaps with other categories, except for the windmill in the center of the latent space in Figure 6. This means windmill can have more "interactions" with other categories. For most categories, the numbers of sketches having overlaps with more than three categories are all below 250%, 10% of the total sketches in each category. In the section "Visual similarity quantification and rank", we consider the overlap within at

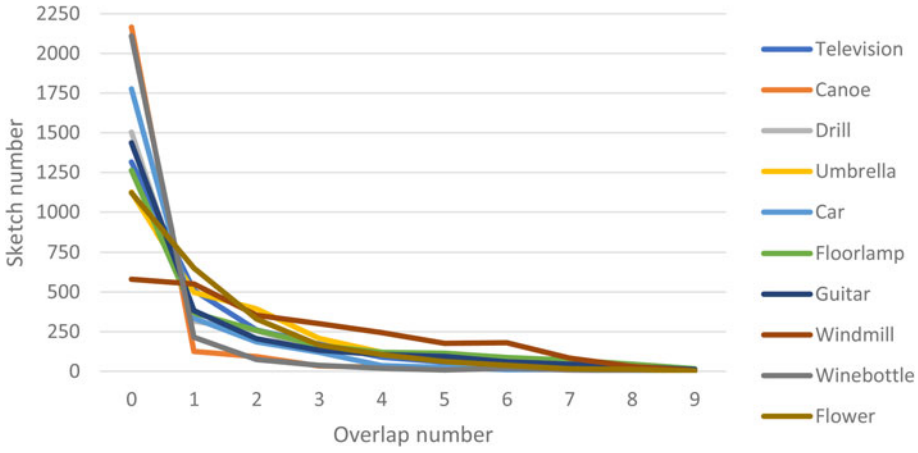


Fig. 7. Sketch number distributions on different overlap numbers for 10 categories.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

most n ($1 \leq n \leq 9$) categories to calculate the weights in the top clustering detection-based method. However, in Figure 6, we only consider the overlap within at most three categories to show that sketches in different regions from the same category can have slightly different shapes. Take the "television" category as an example. As shown in Figure 6, t_1 is a native point. It means t_1 can represent the most frequently drawn television sketch by users. It includes two rectangles to present the monitor and two intersected lines to present an antenna. t_2, t_3, and t_4 are native-overlap (2) points. t_2 is in the overlap region of car and television. It has a long thin rectangle which is like the body of a car. t_3 is in the overlap region of windmill and television. It has an antenna that can be regarded as blades and a base stand which can be regarded as the body of a windmill. t_4 is in the overlap region of wine bottle and television. It has a long tall shape which is like the appearance of a wine bottle. t_5 is a native-overlap (3) point. It is in the overlap region of car, television, and windmill. It has a base stand and a long thin rectangle. t_6 is a departed point. It is "mistakenly" clustered in the car category and not overlapped with other categories. It has two circles which are like two wheels and a long thin rectangle that is like the body of a car. t_6 has more shape features of a car than t_2. t_7 is a departed-overlap (2) point. It is "mistakenly" clustered in the car category and overlapped with canoe category. For t_6 and t_7, it is not so obvious to see they belong to television sketches.

Drill and guitar are separated into several subgroups and located in different areas in the latent space. After retrieving sketches, we can find, for drill, one major cluster can represent handheld drill, another can represent ground drill; For guitar, one major cluster is acoustic, and another major is electric. Other clusters from these two categories have multiple orientations and slight shape modifications. This shows rotation and shape change can affect the results of clustering. Since windmills and flowers share too many shape features, they are merged into each other. In the clockwise or anti-clockwise direction, we can see the shape is gradually changing across different categories. The localization of sketch categories in the latent space suggests that the learned features are very useful for both within-category and cross-category sketch retrievals. Categories with higher visual similarity have shorter distances. The visual similarity should be quantified before we can effectively retrieve sketches.

Visual similarity quantification and rank

In the latent space, when the categories are densely clustered, such as canoe and car, it is easy to use Euclidean distance to measure the similarity between them. However, when categories are mixed together, such as windmill and flower, or separated as several subgroups, such as drill and guitar, Euclidean distance may not be accurate enough. In this section, we compare the proposed top clustering detection (TCD) method with the Euclidean distance-based method to measure the visual similarity between different categories. Let s_{ij} present the similarity between category i and j based on Euclidean distance, which can be computed as follows (Karimi *et al.*, 2019):

$$s_{ij} = 1 - \frac{d_{ij}}{\max_j d_{ij}}, \tag{11}$$

where d_{ij} is the Euclidean distance between category i and j, and $\max_j d_{ij}$ is the longest Euclidean distance from category i to other categories.

Figure 8 visualizes the similarity matrix for 10 categories; the size of a square means the value of similarity. A larger square

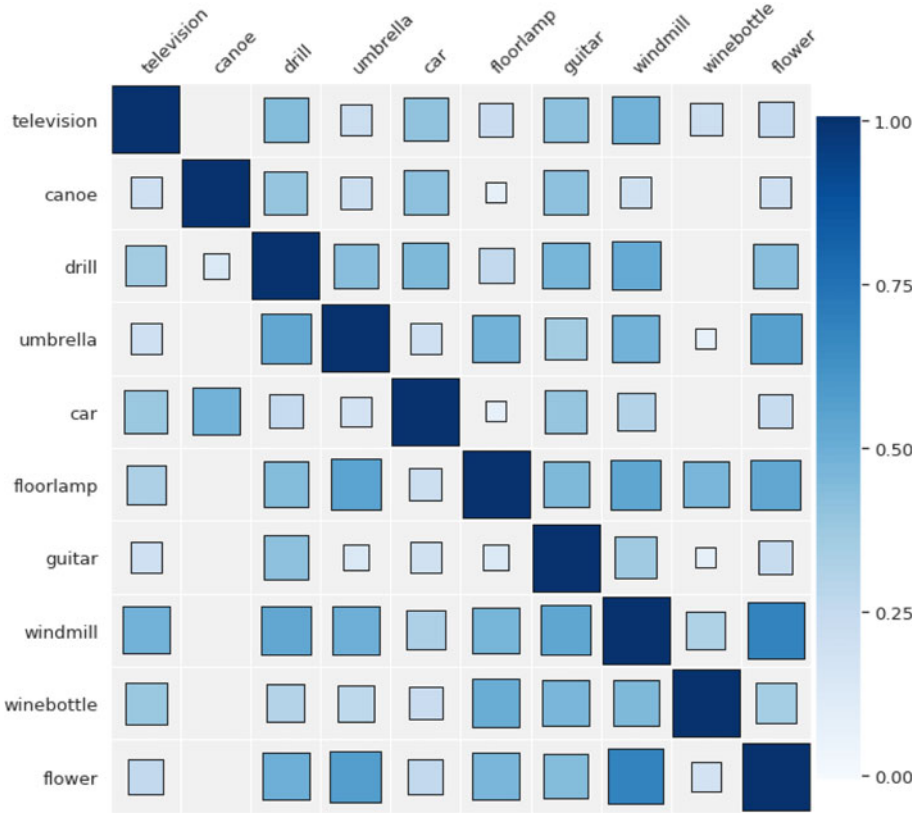


Fig. 8. Similarity matrix based on Euclidean distance.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

means a higher similarity value. For each row, it represents the similarity magnitude of other categories to the row category. It is not easy to find the largest square immediately. It means similarity magnitude is not so differentiable for each category as the several distances are so close. For categories that are separated into some subgroups, such as drill and guitar, using Euclidean distance may not be accurate. Drill is more similar to car than umbrella as its centroid is closer to that of the cars. After checking the sketches, umbrella should be more visually similar to drill. Many points of guitar are merged into the wine bottle cluster. However, the similarity value of guitar to wine bottle is low as the centroid distance between them is large.

Figure 9 visualizes the similarity between different categories based on our top clustering detection method with different overlap values. Overlap value means the largest number of categories one category can have overlaps with when calculating similarity based on the TCD method. For example, if the overlap value equals 1, it means one category is inclined to build a similarity relationship with the nearest one category in the latent space. In Figure 9, when the overlap value becomes larger, it is easier to see that one category has more chance to build similarity relationships with more categories. For example, when the overlap value is 1, we can only find at most one category that is similar to it for each category. When the overlap value is 3 or 8, we can easily find several categories that are similar to it for each category. The reason is when using the TCD method to calculate the similarity values between one category to other categories, the sketches in the overlap regions with more categories are counted when the overlap value becomes larger. How to choose the optimal overlap value is a tradeoff. We need to consider how many visually similar or dissimilar categories should be retrieved and present to the users. If setting the overlap value to a small number, users may miss some important visual similarity information. If setting the overlap value to a large number, the user may be provided too much visual similarity information to filter some significant ones.

In Figure 9, the categories are arranged in the same order as shown in Figure 8 to aid comparison. From Figure 9, it is easier to figure out the visual similarity relationship between different categories than in Figure 8, as every point in the latent space contributes to computing the similarity rather than measuring the distance of the centroids. Take Figure 9(b) as an example to compare with Figure 8. For dense clusters with few overlapped regions with other categories, such as canoe, car, television, floor lamp, and umbrella, two methods have similar rankings for each category. However, for categories that are separated in the latent space, such as drill and guitar, the top clustering detection-based method is more reasonable for the similarity ranking. The distance between drill and windmill is shorter than drill and guitar. Guitar is ranked higher than windmill, which is more reasonable as more points of guitar mixed with drill. There is one subgroup of guitar which is separated away from other subgroups and merges with wine bottle. The centroid of guitar cannot accurately reflect the locations of some sketches in the latent space. Drill has the shortest Euclidean distance category from guitar. However, wine bottle has the largest overlapped region with it. Based on our method, wine bottle is the most similar category to guitar. Canoe has the longest distance from guitar. However, it has the second-largest overlapped region with guitar. Therefore, it is the second similar category in our method and last similar category based on Euclidean distance. Also, the most similar category to wine bottle is guitar rather than the shortest distance category floor lamp. Finally, for each category, the visual similarity rank can be obtained and stores in a database.

Within-category and cross-category retrieval

The retrieval performance of our proposed method is evaluated by top-k retrieval accuracy with majority hits. The k represents the number of nearest neighbors of the query in the latent space. The majority hits mean more than half of k nearest neighbors having exactly the same category label as the query. In the experiment for evaluating the retrieval performance, the label of a query is known beforehand. The criteria to evaluate whether a model has a good retrieval performance is the retrieval accuracy rate which equals the number of successful major hits over the total number of the sample data. The retrieval accuracy rate can reflect the shape feature extraction and clustering performance of the model. In Figure 10(a), the query label is blue, and the major label of the 5 nearest neighbors is also blue. Therefore, it is a successful retrieval with major hits. In Figure 10(b), the query label is blue, and the major label of the 5 nearest neighbors is also green. Therefore, it is an unsuccessful retrieval with major hits, meaning that this query is more visually similar to the green cluster.

We randomly sampled 10 K sketches from each category as queries. These sketches are from the Quickdraw dataset and are not used in training our model. The top-k retrieval accuracy at different k with different hits is plotted in Figure 11. The retrieval accuracy rate can reflect the density of each category in the latent space. For example, as car is the densest category in Figure 6, it

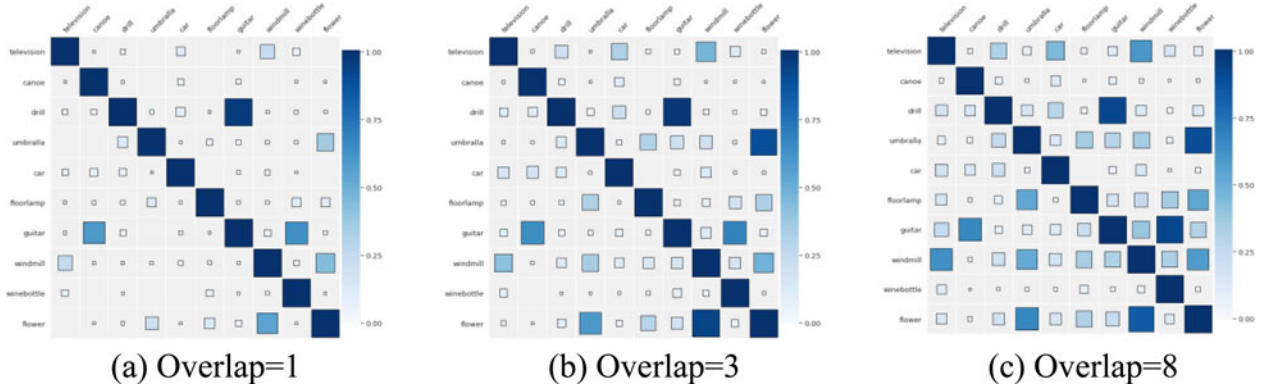


Fig. 9. Similarity matrix based on TCD with different overlap values.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

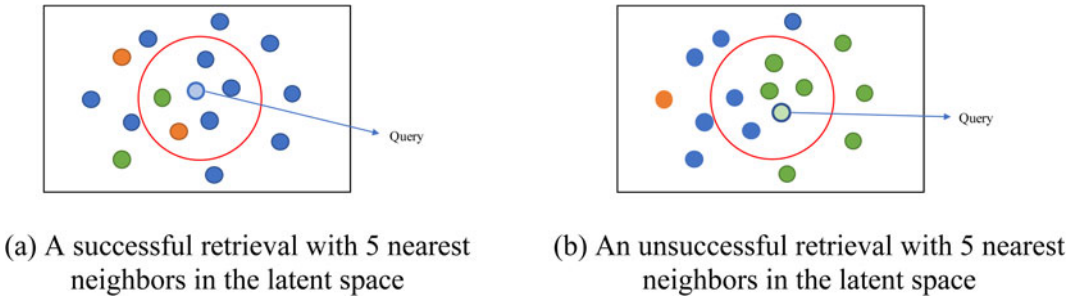


Fig. 10. The successful and unsuccessful retrieval with 5 nearest neighbors in the latent space.

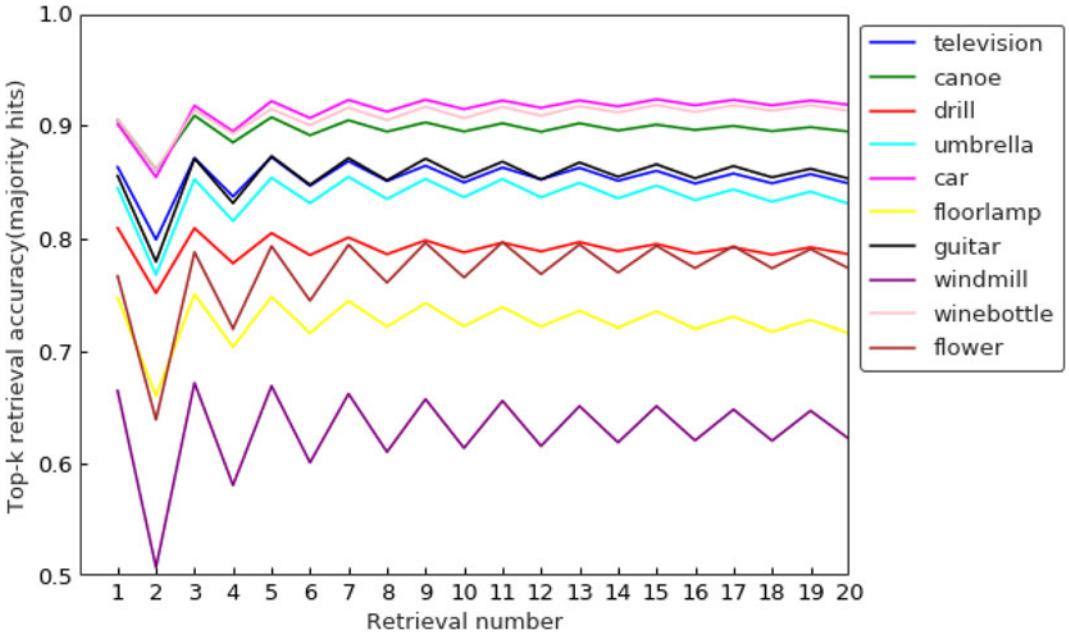


Fig. 11. The top-*k* retrieval accuracy with majority hits of each category.

has the highest top-*k* retrieval accuracy rate with different *k* values. As windmill is the loosest category in Figure 6, it has the lowest top-*k* retrieval accuracy rate with different *k* values. In general, the top-5 retrieval accuracy rate is nearly the highest for all 10 categories. Therefore, we choose 5 nearest neighbors of the query and determine which category the query is most visually similar to.

Our generated visual analogy database quantifies the visual similarity between different sketch categories and stores these visual relationships. In the visual stimuli searching and retrieving scenario, if designers would like to find short-distance and long-distance visual stimuli in our system, firstly, our systems need to know which category has the highest visual similarity with the query. Therefore, the major category label of the top 5 nearest neighbors of the query is determined. In this category, designers can most likely find short-distance visual analogies (e.g., images and category information). We call the major category label the *assigned label*. Also, based on the quantified visual relationships, our system can easily find long-distance visual analogies (e.g., images and category information) from several categories having low similarity with the assigned category. Ten retrieval experiments are done before deciding on the appropriately assigned label for the query. In Table 2, we show some examples of assigning labels to queries based on the top-5 retrievals. We choose five categories and list them in rows based on the top-5 retrieval accuracy in descending order. From Table 2, we can see categories with high top-5 retrieval accuracies, such as car, wine bottle, and television, have a higher possibility of being assigned correct (ground truth) labels than categories with low top-5 retrieval accuracy. The categories with higher retrieval accuracy imply they are more distinguishable from other categories.

Based on the assigned categories, we can retrieve sketches from within-category and cross-category, as shown in Table 3. For each category, all the points are classified into four different datasets. Given the assigned category label from Table 2, we can retrieve sketches from the four datasets of the assigned category and the most or least similar category of the assigned category based on the visual similarity rank. In Table 3, we randomly retrieve one sketch for each dataset; within-category is the assigned category of the query, and cross-category is the category that has the highest/lowest similarity with the assigned category. In Table 3, we only show sketches from cross-categories with the highest similarity.

Finally, we check whether our model can retrieve similar sketches even if the categories of queries are not in the training dataset. These categories are called unseen categories. Here, we

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

Table 2. Examples of assigned labels and top-5 retrievals for queries

Query	Ground truth label	Assigned label	Top-5 retrieval				
	car	car	 car	 car	 car	 car	 car
	winebottle	winebottle	 winebottle	 winebottle	 winebottle	 winebottle	 winebottle
	television	television	 television	 windmill	 television	 television	 television
	floorlamp	umbrella	 floorlamp	 umbrella	 umbrella	 umbrella	 drill
	windmill	flower	 flower	 flower	 floorlamp	 windmill	 flower

randomly select five unseen categories: a fan, screwdriver, parachute, radio, and roller skates. For each of them, we visualize the top-5 retrievals that are most similar to the unseen categories, as shown in Table 4. We can see that the top-5 retrievals are visually correlated with the sketches from unseen categories, and the generalization ability of our model can be very intuitive and explainable.

Discussion

Visual exploration

In Figure 6, we construct a 2D visual presentation for mapping 2500 sketches of each category in the latent space. Most sketches from the same category are close to each other. It means dc-sketch-pix2seq can capture shape features of 10 categories after training. Sketches can be retrieved based on our proposed top clustering detection (TCD) based method. By visually browsing the visual relationship graph, designers can explore the Quickdraw dataset in the latent space that makes explicit how data points are interconnected based on visual similarity. Sketches in each category are classified into four types based on the locations. Most sketches are not merged into other categories. These sketches can be representatives as they can reflect the unique and salient shapes of the category. The sketches around the boundary of two or more categories or sketches located in the "wrong" categories can be useful to measure similarity between categories. These special sketches exist for two reasons: (1) In the Quickdraw application, users draw sketches based on the keywords of a category which can cause diverse visual understanding. For example, when given "guitar" as a keyword, users may draw an electric or acoustic guitar with different orientations. These variances make it possible to discover visual analogies from other categories by our computational methods. (2) The sketches are usually rough and ambiguous. One sketch can be interpreted and represented in many ways. For example, if a sketch presents a stand with some blades on top of it, it can be understood as a flower or a windmill by our deep learning model. Therefore, visual relationships between these two categories can be built. This graph can aid designers in browsing or creating a visually changing path from one category to other categories, which can be helpful for visual imagination. The reason why we choose the Quickdraw dataset is all images in Quickdraw are simple strokes. It would be easier for our model to extract the relationships between different strokes (black and white pixels) than shapes in colorful images. If we would like to provide more meaningful

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

Table 3. Within-category and cross-category retrieval based on the assigned labels of queries

Query	Within-category retrieval				Cross-category retrieval			
	N	D	NO(2)	DO(2)	N	D	NO(2)	DO(2)
								
								
									
									
									

Note: N means native points, D means departed points, NO (2) means native-overlap points within two categories, DO (2) means departed-overlap points within two categories.

design inspirations, our model needs to consume more domain (including geometric and semantic) knowledge. By doing this, the model can be useful to deal with more realistic cases.

Visual similarity quantification

We compare our method with the Euclidean distance-based similarity measurement. The Euclidean distance-based method cannot differentiate similarity magnitude when the centroids of categories are so close to each other. In Figure 4, we can see the distances between the three centroids are very close. However, the overlap region between orange and green points is larger than the region between orange and blue points. Besides, as users have a diverse visual understanding of the same object, one category can be sketched in variant shapes. For example, "guitar" is separated into several subgroups in Figure 6. The diversity in shapes is important to build a visual relationship between different categories. However, this diversity can contribute to the imprecise measurement of similarity when using the Euclidean distance. Our proposed method measures similarity based on the overlapping magnitude between categories, which is more appropriate and accurate to handle the two aforementioned problems. The distance of analogy can be quantitatively measured based on the overlap magnitude. More overlapped regions mean more shape features shared between the categories in the latent space, leading to shorter analogy distance and vice versa. One potential flaw of the TCD method is it would mistakenly classify many sketches into wrong types. For example, many native sketches are wrongly classified into departed sketches. If this happens, the similarity matrix of the TCD method would be wrong. One way to validate the results of the TCD method is to compare them with the similarity matrix based on Euclidean distance. If these two methods have similar results for each category, the similarity measure between categories based on the TCD method is solid. If not, it means the clustering performance of the proposed dc-sketch-pix2seq model is not strong enough. We need to modify the structure of the model to make it stronger.

Sketch retrieval performance

The assigned category of the query is based on the top-k retrieval results. Given the assigned category, our method can retrieve sketches from the same category; Given the visual similarity measures, the most and least similar categories can be decided. Sketches in these categories can be retrieved as cross-category visual stimuli. Because of this cross-category retrieving capability, our method can potentially help expand designers' visual thinking limits. The sketches from the most and least similar categories can be regarded as short- and long-distance visual analogies. Besides, we have found if a category has a low top-k retrieval accuracy, visually similar sketches from other categories can be easily retrieved. The reason is that this category has larger overlap regions with other categories, or it has multiple subgroups connected with other categories. Therefore, we can detect windmill and car are the most possible and impossible categories to build visual relationships with other categories, respectively. Fu *et al.* have shown that a long-distance is not necessarily desirable for ideation (Fu *et al.*, 2013*b*). The long-distance here means the stimuli are too far, and they then can become harmful to the design process. "Near" and "far" when talking about the distance of analogies often mean something different to each researcher and to each individual study or discussion. In this paper, we do

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

Table 4. Top-5 most similar sketches to the queries from unseen categories

Queries from unseen categories	Top-5 retrieval				
					
fan	windmill	windmill	flower	windmill	flower
					
screwdriver	guitar	guitar	guitar	guitar	guitar
					
parachute	umbrella	umbrella	umbrella	umbrella	flower
					
radio	television	television	television	television	television
					
roller skates	car	car	car	television	car

not explicitly quantify short- and long-distance. The sketches from the most and least similar categories can be regarded as short- and long-distance visual analogies. We have another paper that discusses how to quantify short- and long-distance analogy (Zhang and Jin, 2021). In that paper, we also have some experiments to show how short- and long-distance analogy can be found and related to the "sweet spot", which was put forward by Fu *et al.* (Fu *et al.*, 2013*b*).

Conclusion

In this paper, we developed the dc-sketch-pix2seq model to learn shape feature presentations and proposed a TCD-based method to quantify visual similarities of the learned representations of sketches. They work together and form a visual stimuli search and retrieval framework, which provides external visual cues for designers to potentially enhance their visual analogy capabilities. In summary, the main contributions of this paper are:

1. A computational framework is proposed to automatically search and retrieve sketches from *various* categories based on *quantified* visual similarity, which has been lacking in the area of design by visual analogy support. Currently, most sketch-based image retrieval (SBIR) tools focus on accurately retrieving relevant sketches which are in the same category as the query sketch. The proposed dc-sketch-pix2seq model clusters shape features during training the deep learning model, which ensures retrieving sketches from the *same* or *different* categories following the query and enables visual exploration.
2. The performance of search and retrieval functions of our proposed computational framework has been evaluated on a large sketch dataset as design source materials, which stands for a novel and meaningful exploration of visual data-enabled design support. In the area of design by analogy, little research has been done to allow exploration into large image datasets from different domains, and our computationally evaluated model can be utilized as a tool to search and retrieve more meaningful visual analogies in human-based design by analogy studies.

In future work, we will have two main extensions. First, visual similarity may not always indicate the presence of a visual analogy. For example, suppose users provide a query image of a palm tree. In that case, they are probably more interested in semantically similar images of other trees such as oaks and maples than in images of spiders. But the images of spiders may be more likely to cause design creativity. Therefore, if the retrieved object is too semantically similar to the query image, it may cause design fixation. If the retrieved object is too visually similar to each other, it may cause irrelevant retrieval. For the experiments in this paper, the sketch categories are the only

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

semantic connections. So, the semantically irrelevant retrievals rarely happened. In our future work, we plan to enrich semantic connections and combine the semantic similarity and the visual similarity to retrieve more meaningful and inspirational stimuli for visual analogy making based on large datasets. Second, the proposed computational framework lacks human validation in an engineering design scenario. Currently, there are no engineering design image datasets available for us to train our model and test how this framework can take in sketches from designers and search and retrieve visual stimuli (engineering designs) to inspire designers to create more design concepts. However, the merit of this research is we know we can follow this framework to search and retrieve visually similar shapes from various domains given the query based on a large sketch dataset. And some qualitative and quantitative evaluations are done to test whether the framework can be workable. If the engineering design image dataset can be built in the future, we can directly implement this framework. The long-term goal of our research is to develop a computational tool that allows designers to perform a visual or shape-based search that can return relevant visual cues to help designers' analogy-making. Human validation is the next step. The validation here refers that we need to recruit some students and designers to (1) use or not use our tools and (2) use or not use certain similarity metrics to search for visual stimuli and track and compare whether they can finally generate some novel designs through visual analogy. Such human–computer interaction experiment results will inform us about the adequacies of both the tool and the similarity metrics.

References

Atilola O, Tomko M and Linsey JS (2016) The effects of representation on idea generation and design fixation: a study comparing sketches and function trees. *Design Studies* **42**, 110–136.

Bogers S, Frens J, van Kollenburg J, Deckers E and Hummels C (2016) Connected baby bottle: a design case study towards a framework for data-enabled design. *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, pp. 301–311.

Cao X, Zhang H, Liu S, Guo X and Lin L (2013) Sym-fish: a symmetry-aware flip invariant sketch histogram shape descriptor. *Proceedings of the IEEE International Conference on Computer Vision*, pp. 313–320.

Casakin H (2010) Visual analogy, visual displays, and the nature of design problems: the effect of expertise. *Environment and Planning B: Planning and Design* **37**, 170–188.

Chakrabarti A, Sarkar P, Leelavathamma B and Nataraju B (2005) A functional representation for aiding biomimetic and artificial inspiration of new ideas. *AI EDAM* **19**, 113–132.

Chakrabarti A, Siddharth L, Dinakar M, Panda M, Palegar N and Keshwani S (2017) Idea Inspire 3.0—a tool for analogical design. *International Conference on Research into Design*. Springer, pp. 475–485.

Chen T, Cheng M-M, Tan P, Shamir A and Hu S-M (2009) Sketch2photo: internet image montage. *ACM Transactions on Graphics (TOG)* **28**, 124.

Chen Y, Tu S, Yi Y and Xu L (2017) Sketch-pix2seq: a model to generate sketches of multiple categories. *arXiv preprint* arXiv:1709.04121.

Cheong H, Chiu I, Shu L, Stone RB and McAdams DA (2011) Biologically meaningful keywords for functional terms of the functional basis. *Journal of Mechanical Design* **133**, 021007.

Chiu I and Shu L (2007) Biomimetic design through natural language analysis to facilitate cross-domain information retrieval. *AI EDAM* **21**, 45–59.

Christensen BT and Schunn CD (2007) The relationship of analogical distance to analogical function and preinventive structure: the case of engineering design. *Memory & Cognition* **35**, 29–38.

Du P and MacDonald EF (2015) Products' shared visual features do not cancel in consumer decisions. *Journal of Mechanical Design* **137**, 071409.

Eitz M, Hildebrand K, Boubekeur T and Alexa M (2010) An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics* **34**, 482–498.

Fu K, Cagan J, Kotovsky K and Wood K (2013*a*) Discovering structure in design databases through functional and surface based mapping. *Journal of Mechanical Design* **135**, 031006.

Fu K, Chan J, Cagan J, Kotovsky K, Schunn C and Wood K (2013*b*) The meaning of "near" and "far": the impact of structuring design databases and the effect of distance of analogy on design output. *Journal of Mechanical Design* **135**, 021007.

Goel V (1995) *Sketches of Thought*. Cambridge, MA: MIT Press.

Goel AK, Vattam S, Wiltgen B and Helms M (2012) Cognitive, collaborative, conceptual and creative—four characteristics of the next generation of knowledge-based CAD systems: a study in biologically inspired design. *Computer-Aided Design* **44**, 879–900.

Goldschmidt G (1994) On visual design thinking: the vis kids of architecture. *Design Studies* **15**, 158–174.

Goldschmidt G (2001) Visual analogy: A strategy for design reasoning and learning. In Eastman CM, McCracken WM and Newstetter WC (eds), *Design Knowing and Learning: Cognition in Design Education*. Amsterdan: Elsevier, pp. 199–220.

Goldschmidt G and Smolkov M (2006) Variances in the impact of visual stimuli on design problem solving performance. *Design Studies* **27**, 549–569.

Gonçalves M, Cardoso C and Badke-Schaub P (2014) What inspires designers? Preferences on inspirational approaches during idea generation. *Design Studies* **35**, 29–53.

Goucher-Lambert K, Moss J and Cagan J (2019) A neuroimaging investigation of design ideation with and without inspirational stimuli—understanding the meaning of near and far stimuli. *Design Studies* **60**, 1–38.

Goucher-Lambert K, Gyory JT, Kotovsky K and Cagan J (2020) Adaptive inspirational design stimuli: using design output to computationally search for stimuli that impact concept generation. *ASME Journal of Mechanical Design* **142**, 091401.

Ha D and Eck D (2017) A neural representation of sketch drawings. *arXiv preprint* arXiv:1704.03477.

Han J, Shi F, Chen L and Childs PR (2018) A computational tool for creative idea generation based on analogical reasoning and ontology. *AI EDAM* **32**, 462–477.

Herring SR, Chang C-C, Krantzler J and Bailey BP (2009) Getting inspired!: understanding how and why examples are used in creative design practice. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, pp. 87–96.

Hu R, Barnard M and Collomosse J (2010) Gradient field descriptor for sketch based retrieval and localization. *2010 IEEE International Conference on Image Processing*, IEEE, pp. 1025–1028.

Jin Y and Benami O (2010) Creative patterns and stimulation in conceptual design. *AI EDAM* **24**, 191–209.

Jongejan J, Rowley H, Kawashima T, Kim J and Fox-Gieg N (2016) The Quick, Draw! - A.I. Experiment.

Karimi P, Maher ML, Davis N and Grace K (2019) Deep learning in a computational model for conceptual shifts in a co-creative design system. *arXiv preprint* arXiv:1906.10188.

Kingma DP and Welling M (2013) Auto-encoding variational bayes. *arXiv preprint* arXiv:1312.6114.

Kuhn HW (1955) The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2**, 83–97.

Kwon E, Pehlken A, Thoben K-D, Bazylak A and Shu LH (2019) Visual similarity to aid alternative-use concept generation for retired wind-turbine blades. *Journal of Mechanical Design* **141**, 031106.

Linsey JS, Clauss E, Kurtoglu T, Murphy J, Wood K and Markman A (2011) An experimental study of group idea generation techniques: understanding the roles of idea representation and viewing methods. *Journal of Mechanical Design* **133**, 031008.

Luo J, Yan B and Wood K (2017) InnoGPS for data-driven exploration of design opportunities and directions: the case of Google driverless car project. *Journal of Mechanical Design* **139**, 111416.

Maaten LVD and Hinton G (2008) Visualizing data using t-SNE. *Journal of Machine Learning Research* **9**, 2579–2605.

Macomber B and Yang M (2011) The role of sketch finish and style in user responses to early stage design concepts. *International Design Engineering*

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press

Technical Conferences and Computers and Information in Engineering Conference, vol. 54860, pp. 567–576.

McKoy FL, Vargas-Hernández N, Summers JD and Shah JJ (2001) Influence of design representation on effectiveness of idea generation. *Proceedings of ASME DETC*, Pittsburgh, PA, September 9–12.

Sarkar P and Chakrabarti A (2008) The effect of representation of triggers on design outcomes. *AI EDAM* **22**, 101–116.

Setchi R and Bouchard C (2010) In search of design inspiration: a semantic-based approach. *Journal of Computing and Information Science in Engineering* **10**, 031006.

Shah JJ, Vargas-Hernandez N, Summers JD and Kulkarni S (2001) Collaborative sketching (C-sketch)—an idea generation technique for engineering design. *The Journal of Creative Behavior* **35**, 168–198.

Shu L (2010) A natural-language approach to biomimetic design. *AI EDAM: Artificial Intelligence for Engineering Design, Analysis, and Manufacturing* **24**, 507–519.

Vattam S, Wiltgen B, Helms M, Goel AK and Yen J (2011) DANE: fostering creativity in and through biologically inspired design. *Design Creativity 2010*. Springer, pp. 115–122.

Walther DB, Chai B, Caddigan E, Beck DM and Fei-Fei L (2011) Simple line drawings suffice for functional MRI decoding of natural scene categories. *Proceedings of the National Academy of Sciences* **108**, 9661–9666.

Yang MC (2009) Observations on concept generation and sketching in engineering design. *Research in Engineering Design* **20**, 1–11.

Yu Q, Liu F, Song Y-Z, Xiang T, Hospedales TM and Loy C-C (2016) Sketch me that shoe. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 799–807.

Zhang Z and Jin Y (2020) An unsupervised deep learning model to discover visual similarity between sketches for visual analogy support. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 83976. American Society of Mechanical Engineers, p. V008T008A003.

Zhang Z and Jin Y (2021) Toward computer aided visual analogy support (CAVAS): augment designers through deep learning. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 85420. American Society of Mechanical Engineers, p. V006T006A057.

Zijian Zhang is a doctoral candidate in the department of Aerospace & Mechanical Engineering at the University of Southern California. His research covers design theory & methods, deep learning and machine learning. His current research is to apply deep learning to develop a computer aided visual analogy support (CAVAS) framework to support engineering design. Mr. Zhang is a recipient of the 2021 ASME Design Theory and Methodology Best Paper Award.

Yan Jin is Professor of Aerospace and Mechanical Engineering at the University of Southern California. He received his PhD degree in Naval Engineering from the University of Tokyo and did his post-doctoral research at Stanford University. Dr. Jin's research covers design theory & methods, multiagent, and self-organizing systems. His current research interests include design cognition, machine learning and its applications in engineering design, knowledge capturing, self-organizing, and adaptive systems. Dr. Jin is a recipient of the National Science Foundation CAREER Award and TRW Excellence in Teaching Award. He served as Editor-in-Chief of AIEDAM and Associate Editor of JMD and is currently Associate Editor of Design Science Journal. Dr. Jin is a Fellow of ASME.

https://doi.org/10.1017/S0890060422000063 Published online by Cambridge University Press