Towards Deep Universal Sketch Perceptual Grouper

Ke Li[1,2] Kaiyue Pang[2] Yi-Zhe Song[2] Tao Xiang[2] Timothy M. Hospedales[2,3] Honggang Zhang[1]
[1]Beijing University of Posts and Telecommunications [2]SketchX, Queen Mary University of London
[3]The University of Edinburgh

***Abstract*—Human free-hand sketches provide useful data for studying human perceptual grouping, where the grouping principles such as the Gestalt laws of grouping are naturally in play during both the perception and sketching stages. In this work, we make the first attempt to develop a universal sketch perceptual grouper. That is, a grouper that can be applied to sketches of any category created with any drawing style and ability, to group constituent strokes/segments into semantically meaningful object parts. The first obstacle to achieving this goal is the lack of large-scale datasets with grouping annotation. To overcome this, we contribute the largest sketch perceptual grouping (SPG) dataset to date, consisting of $20,000$ unique sketches evenly distributed over 25 object categories. Furthermore, we propose a novel deep perceptual grouping model learned with both generative and discriminative losses. The generative loss improves the generalisation ability of the model, while the discriminative loss guarantees both local and global grouping consistency. Extensive experiments demonstrate that the proposed grouper significantly outperforms the state-of-the-art competitors. Additionally, we show that our grouper is useful for a number of sketch analysis tasks including sketch semantic segmentation, synthesis and fine-grained sketch-based image retrieval (FG-SBIR).**

***Index Terms*—Sketch Perceptual Grouping, Universal grouper, Deep grouping model, Dataset.**

I. Introduction

Humans effortlessly detect objects and object parts out of a cluttered background. The Gestalt school of psychologists [1], [2] argued that this ability to perceptually group visual cues/patterns into objects (parts) is built upon a number of grouping principles, termed Gestalt laws of grouping. These include five laws, namely proximity, similarity, continuity, closure, and symmetry [3], which have long been exploited by the computer vision researchers studying grouping or segmentation. For example, in image segmentation [4], [5], [6], [7], pixel visual appearance similarity and local proximity are often used to group pixels into objects/parts. Exploiting these principles has been beneficial because these are the grouping strategies used by the human visual system in diverse contexts and for diverse object categories. Exploiting them, either explicitly or implicitly, is thus also likely to be useful for developing a universal grouping algorithm.

We aim to develop such a grouper for human free-hand sketches which takes a sketch as input and groups the constituent strokes into semantic parts. Note that this is different from semantic segmentation for either photos [5] or sketches [8], [9], [10], where each segmented part is given a label, and the labels are often object category-dependent (*e.g.*, nose of a face and wings of an aeroplane). In our problem, only the group relationship between strokes is predicted so that the grouper can universally be applied to any object category (*i.e.*, we only care about whether two strokes belong to the same group, not which object part the group corresponds to). Human free-hand sketches provide an ideal testbed for applying/evaluating algorithms developed to exploit the human perceptual grouping principles. This is because they are drawn by humans to reflect their perception of visual objects and their parts. The grouping principles are thus in play during both the perception and sketching stages.

Although sketch perceptual grouping is an interesting problem on its own and has potential to benefit related areas such as more general image segmentation and psychophysics, very few works exist [11], [12]. These approaches typically compute hand-crafted features from each stroke and use the proximity and continuity principles to compute a stroke affinity matrix for subsequent clustering/grouping. They thus have a number of limitations: (i) Only two out of the five principles are exploited, while the unused ones such as closure are clearly useful in grouping human sketches which can be fragmented (see Fig. 1). (ii) How the principles are formulated is determined manually rather than learned from data. (iii) Fixed weightings of different principles are used which are either manually set [11] or learned [12]. However, for different sketches, different principles could be used by humans with different weightings. Therefore a more dynamic sketch-specific grouping strategy is preferable. To overcome these limitations, a data-driven approach is more appropriate, by which different principles used by the human sketchers and their weightings are automatically discovered from data. Nevertheless, the existing sketch perceptual grouping datasets [8], [12] are extremely small, containing 2,000 annotated sketches at most. This hinders the development of a data-driven approaches, especially those based on powerful and flexible deep neural network models.

The first contribution of this paper is to provide the first large-scale sketch perceptual grouping (SPG) dataset consisting of 20,000 sketches with ground truth grouping annotation, *i.e.*, 10 times larger than the largest dataset to date [12]. The sketches are collected from 25 representative object categories with 800 sketches per category. Some examples of the sketches and their annotation are shown in Fig. 1. A dataset of such size makes the development of a deep universal grouper possible.

Even with sufficient training samples, learning a deep universal sketch grouper is non-trivial. In particular, there are two main challenges: how to make a deep grouper generalise to unseen object categories and domains/datasets without any training data from them; and how to design training losses that enforce both local (stroke pairwise) grouping consistency and global (whole sketch level) grouping consistency given

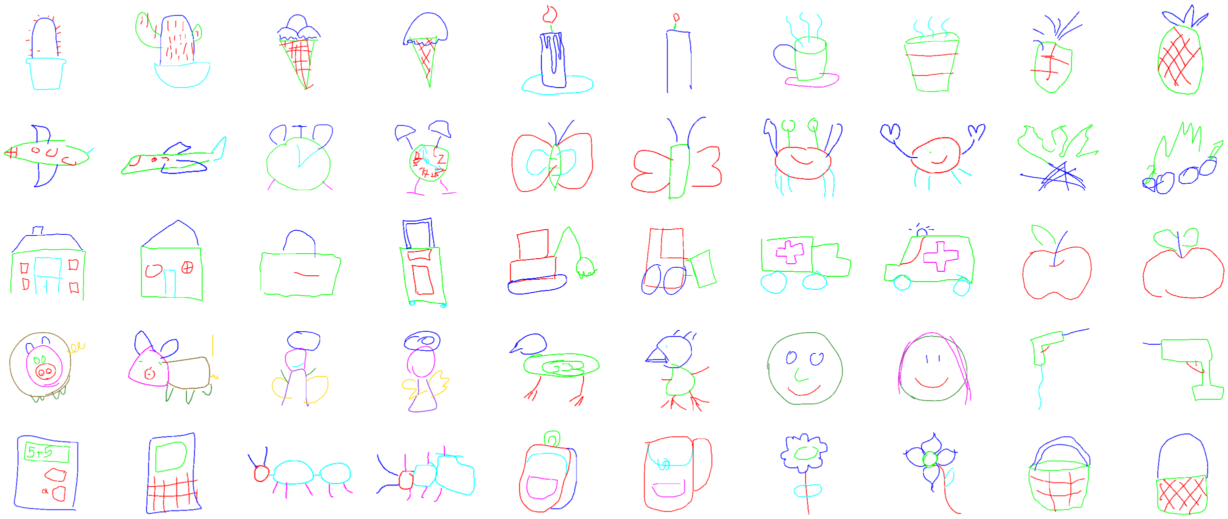


Fig. 1: Examples of the SPG dataset. Stroke groups are colour coded.

variable number of strokes per sketch. Most losses used by existing deep models are for supervised classification tasks; grouping is closer to clustering than classification, so few options exist.

In this paper, we propose a novel deep sketch grouping model to overcome both challenges. Specifically, treating a sketch as a sequence of strokes/segments, our model is a sequence-to-sequence variational auto-encoder (VAE). The reconstruction loss in this deep generative model forces the learned representation to preserve information richer than required for the discriminative grouping task alone. This has been proven to be useful for improving model generalisation ability [13], critical for making the grouper universal. As for the discriminative grouping learning objectives, we set out for two goals: (i) a pairwise stroke grouping loss enforcing local grouping consistency; (ii) both hard triplet ranking and instance-level centre losses to enforce global grouping consistency. This separation of the local and global grouping losses enables us to balance the two and makes our model more robust against annotation noise.

Using our grouping model, we not only have a universal grouper that groups strokes into objects parts, but also armed to address a number of related problems: (i) **Sketch segmentation:** We can repurpose our model for sketch segmentation task by incorporating additional supervised classification loss, *i.e.*, each sketch segment is classified from a pre-defined list of per-category semantics in the form of part labels. (ii) **Instance-level photo-to-sketch synthesis:** Given a photo, we extract an edgemap and treat it as a sketch with extremely fine detail. Our proposed grouper is then applied to abstract the edgemap into a more abstract sketch by first grouping the edges followed by removing the least prominent groups. Note that we do not attempt to explicitly model the human sketch style of abstraction as in [14], [15]. (iii) **Unsupervised Fine-grained SBIR (FG-SBIR):** The synthesis model above is used to synthesise photo-freehand sketch pairs using photo input only. This allows us to by-pass the expensive photo-sketch pair collection step and train an *unsupervised* FG-SBIR model.

Our contributions are as follows: (1) We contribute the largest sketch perceptual grouping dataset to date with extensive human annotation. The dataset is made publicly available at https://github.com/KeLi-SketchX/SketchX-PRIS-Dataset. (2) For the first time, a deep universal sketch grouper is developed based on a novel deep sequence-to-sequence VAE with both generative and discriminative losses. (3) Extensive experiments show the superiority of our grouper against existing ones, especially when evaluated on new categories or new dataset domains. Its usefulness on a number of sketch analysis tasks including sketch segmentation, sketch synthesis and FG-SBIR is also demonstrated.

II. Related work

Perceptual Grouping: Humans can easily extract salient visual structure buried in background clutter and noise. Gestalt psychologists referred to this phenomenon as perceptual organisation [1], [2] and introduced the concept of perceptual grouping, which accounts for the observation that humans naturally group visual patterns into objects. A set of simple Gestalt principles were further developed, including proximity, similarity and continuity [3], with closure, connectedness and common fate introduced later, primarily for studying human vision systems [4], [16].

Sketch Groupers: Very few studies exist on grouping sketch strokes into parts. The most related studies are [11], [12]. They compute an affinity matrix between strokes using hand-crafted features based on proximity and continuity principles. The two principles are combined with fixed weights learned from human annotated stroke groups. In contrast, we assume that when humans draw sketches and annotate them into groups, all grouping principles could be used. Importantly, using which ones and by how much are dependent on the specific sketch instance. Our model is thus a deep neural network that takes the sketch as input and aims to model all principles implicitly via both generative and discriminative grouping losses. Consequently, it has the potential to perform principle selection and weighting dynamically according to a given sketch input. We also provide a much larger dataset compared to the one provided in [12]. We show that on both datasets, our model outperforms that in [12] by a big margin. Note that perceptual grouping has been modelled for photo images using a deep autoencoder in [17]. However, the objective is to group discrete graphical patterns which has richer visual cues that make them more akin to the problem of image segmentation, and thus easier than grouping line drawings in sketches.

Sketch Semantic Segmentation: A closely related problem to sketch grouping is sketch semantic segmentation [8], [9], [10][1]. The key difference is that a sketch grouper is universal in that it can be applied to any object category as it only predicts whether strokes belong to the same group rather than what group. In contrast, sketch segmentation models need to predict the label of each group. As a result, typically one model is needed for each object category. Note that although two different problems are tackled, our work can be potentially repurposed for sketch semantic segmentation task since our SPG dataset also contains group ID labels for each category, *e.g.*, by modifying/fine-tuning our model to a fully supervised one. In this work, we show that our proposed perceptual grouping constraints is beneficial for the semantic segmentation task when adopted as a pretraining step.

Sketch Stroke Analysis: Like our model, a number of recent sketch models are based on stroke modelling. [10] studied stroke semantic segmentation. A sequence-to-sequence variational autoencoder is used in [18] for a different purpose of conditional sketch synthesis. The work in [19] uses a sketch RNN for sketch abstraction problem by sequentially removing redundant strokes. A stroke-based model is naturally suited for perceptual grouping – modelling Gestalt principles is harder if a sketch is treated as a 2D pixel array instead of strokes.

Fine-grained SBIR: FG-SBIR has been a recent focus in sketch analysis [20], [21], [22], [23], [24], [25]. Training a FG-SBIR model typically requires expensive photo-sketch pair collection, which severely restricts its applicability to large number of object categories. In this work, we show that our universal grouper is general enough to be applied to edgemaps computed from object photos. The edgemaps can then be abstracted by removing the least important groups. The abstracted edgemap can be used to substitute human sketches and form synthetic sketch-photo pairs for training a FG-SBIR model. We show that the performance of a model trained in this way approaches that of the same model trained with human labelled data, and is superior to the state-of-the-art unsupervised alternative [19].

An earlier and preliminary version of this work was published in [26]. Compared with [26], apart from more extensive experiments and analysis, this work differs in that (i) the global grouping loss is now formulated with two parts: a segment-level loss based on triplet ranking and a group-level one based on centre loss. These changes brings clear improvements in model performance. (ii) our model is generalised for semantic segmentation and shown to produce superior performance compared with the existing sketch segmentation methods.

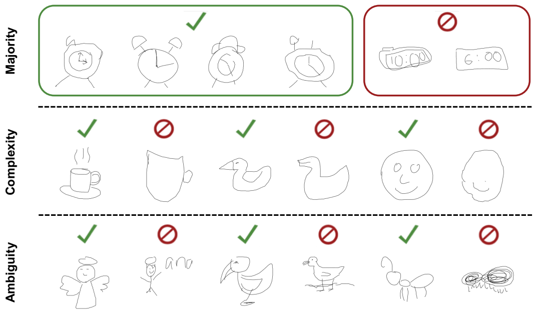


Fig. 2: Examples to illustrate our sketch selection process. See details in text.

III. Sketch Perceptual Grouping Dataset

We contribute the Sketch Perceptual Grouping (SPG) dataset, the largest free-hand sketch perceptual grouping dataset to date. It contains $20,000$ sketches distributed over 25 categories with each sketch manually annotated into parts.

Category Selection: The sketches come from the QuickDraw dataset [18], which is by far the largest free-hand sketch dataset. It contains 345 categories of everyday objects. Out of these, 25 are selected for SPG based on following criteria: (i) Complexity: the category should contain at least three semantic parts, meaning categories such as cloud and moon are out. (ii) Variety: The selected categories need to be sufficiently different from each other to be appropriate for testing the grouper's generalisation ability to unseen classes. For example, only one of the four-legged animal classes is chosen. A full list of SPG categories can be found in Table I and II.

Sketch Instance Selection: Each QuickDraw category contains at least 100,000 sketches. Although it is desirable to annote the entire dataset, the amount of manual annotation required would be impractical for the purpose of this work. So 800 sketches are chosen from each category. First, some quality screening is performed. Specifically, since all QuickDraw sketches were drawn within 20 seconds, there are a large number of badly drawn sketches that are unrecognisable by humans, making part grouping impossible. We thus first discard sketches which could not be recognised by an off-the-shelf sketch classifier [27]. The remaining sketches are then subject to the following instance selection criteria: (i) **Majority**: Sketches in each category often form subcategories which can be visually very different from each other. Only the sketches from the majority subcategory are selected, *e.g.*, the top row of Fig. 2 shows that most sketches from the alarm clock category belong to the "with hand" subcategory, whilst a small minority depicts digital clocks without hands. Only sketches from the former are selected. (ii) **Complexity:** Over-abstract sketches with less than three parts are removed. (iii) **Ambiguity:** We eliminate sketches that contain both the target object and other objects/background to avoid ambiguity of the

[1] Their relationship is analogous to that between unsupervised image segmentation [6], [7] and semantic segmentation [5].

object category. Examples of how these criteria are enforced during instance selection can be seen in Fig. 2.

Annotation: After the collection process, we recruited 25 annotators and asked a single annotator to label an entire category. Each annotator is then required to first go through the assigned category to obtain a rough perceptual understanding of the category-level diversity and complexity, and then to define a taxonomy of group IDs for all the semantic parts he/she believes to be essential for each given category. The additional requirement of group ID annotation has two benefits: (1) it helps produce global and cross-instance consistent grouping annotation, and (2) it means that our SPG dataset can also be used for sketch semantic segmentation. Examples of the annotation can be seen in Fig. 1.

It is noteworthy that the above annotation procedure design is very different from previous semantic segmentation dataset annotation processes, where each data sample is annotated by multiple annotators and majority vote is typically used to deal with label ambiguity. The reason we took a different annotation approach is because our problem is different: the dataset is designed to learn a universal grouper, *i.e.*, generalisable across categories and human drawers. Had this been done by multiple annotators, it would introduce an averaging effect, making the dataset less suitable for evaluating the across-person generalisability of a grouper.

IV. Methodology

A. Model Overview

Our deep sketch grouper is a variant of the sequence-to-sequence variational auto-encoder (VAE) [28], [29]. As shown in Fig. 3, it is essentially a deep encoder-decoder with both the encoder and decoder being RNNs for modelling a sketch as a set of strokes. The encoder produces a global representation of the sketch, which is used as a condition for a variational decoder that aims to reconstruct the input sketch. Note that sketch synthesis is only a side task here. Our main aim is for the decoder to produce a representation of each stroke useful for grouping them. Once learned, the decoder should implicitly model all the grouping principles used by the annotators in producing the grouping labels, so that the learned stroke representation can be used to compute a stroke affinity matrix indicating the correct stroke grouping. To this end, the decoder has two branches: a generative branch to reconstruct the input sketch; and a discriminative branch that produces the discriminative stroke feature/affinity matrix.

B. Encoder and Decoder Architecture

Traditional perceptual grouping methods treat sketches as images composed of static pixels, thus neglecting the dependency between different segments and strokes (each stroke consists of a variable number of line segments). In our dataset, all the sketches are captured in a vectorised format, making sequential modelling of sketches possible. More specifically, we first represent a sketch as a sequence of N stroke-segments $[S_1, S_2, ..., S_N]$. Each segment is a tuple $(\Delta x, \Delta y, p)$, where Δx and Δy denote the offsets along the horizontal and vertical directions respectively, while p represents the drawing state, following the same representation used for human handwriting [30].

With these stroke segments as inputs, both the encoder and decoder are RNNs. In particular, we adopt the same architecture as in Sketch-RNN [18] for conditional sketch synthesis. That is, a bi-directional RNN [31] is used as the encoder to extract the global embedding of the input sketch. The final state output of the encoder is then projected to a mean and a variance vector, to define an IID Gaussian distribution. That distribution is then sampled to produce a random vector z as the representation of the input sketch. Thus z is not a deterministic output of the encoder given a sketch, but a random vector conditional on the input. The decoder is an LSTM model. Its initial state is conditional on z via a single fully connected (FC) layer. At each time step, it then predicts the offset for each stroke segment in order to reconstruct the input sketch. For further details on the encoder/decoder architecture, please refer to [18].

C. Formulation

The decoder splits into two branches after the LSTM hidden cell outputs: a generative branch to synthesise a sketch and a discriminative branch for grouping. Different learning objectives are used for the two branches: in the generative branch, two losses encourage the model to reconstruct the input sketch; in the discriminative branch, the sketch grouping annotation is used to train the decoder to produce an accurate stroke affinity matrix for grouping.

Group Affinity Matrix: The grouping annotation is represented as a sparse matrix denoting the group relationship between segments $\mathbf{G} \in \mathbb{R}^{N \times N}$. Denoting the i^{th} sketch segment as $S_i, i \in [1, N]$, we have:

$$G_{i,j} = \begin{cases} 1, & if\ S_i, S_j\ are\ from\ the\ same\ group \\ 0, & otherwise \end{cases} \tag{1}$$

where each element of the matrix indicates whether the i^{th} and j^{th} segments belong to the same group or not. A straightforward design of the discriminative learning objective is to make the affinity matrix computed using the learned stroke feature $f_i = \phi(S_i)$ as similar as possible to $\mathbf{G}$, via an l_1 or l_2 loss. However, we found that in practice this works very poorly. This is because $\mathbf{G}$ conveys two types of grouping constraints: each element enforces a binary pairwise constraint for two segments, whilst the whole matrix also enforces global grouping constraint, *e.g.*, if S_1 and S_2 are in the same group, and S_2 and S_5 are also in the same group, then global grouping consistency dictates that S_1 and S_5 must also belong to the same group. Balancing these two is critical because pairwise grouping predictions are typically noisy and can lead to global grouping inconsistency. However, using a single loss makes it impossible to achieve a balance. We thus propose to use two types of losses to implement the two constraints.

Local Grouping Loss: This loss requires that the pairwise relationship between two segments are kept when the pairwise affinity is measured using the learned stroke segment feature. The decoder LSTM learns a mapping function ϕ and map

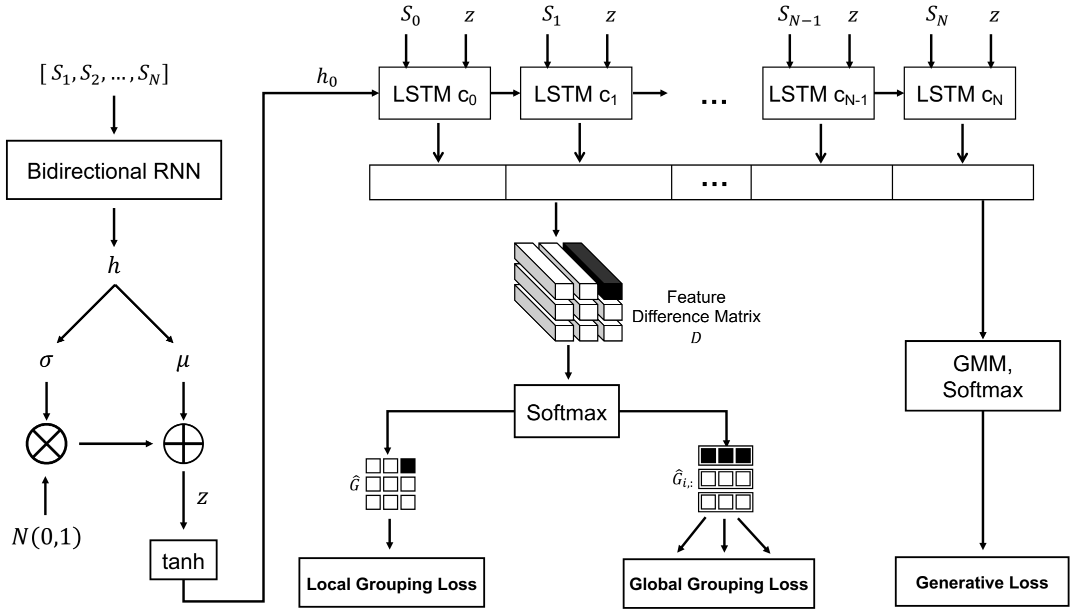


Fig. 3: A schematic of the proposed deep perceptual grouper.

the i^{th} stroke segment S_i to a 128D feature vector f_i. To measure the affinity of any two segments in the input sketch, the absolute element-wise feature difference is computed to obtain a symmetric absolute feature difference matrix, $\mathbf{D} \in \mathbb{R}^{\mathbf{N}\times\mathbf{N}\times\mathbf{128}}$ as:

$$\mathbf{D} = \big\{D_{i,j} \bigm| i,j \in [1,N]\big\} = \big\{|f_i - f_j| \bigm| i,j \in [1,N]\big\} \quad (2)$$

Each vector $D_{i,j} \in \mathbb{R}^{128}$ is then subject to a binary classification loss (cross-entropy) to obtain the local affinity prediction $\hat{G}_{i,j}$, between the i^{th} and j^{th} segments. The local grouping loss, $\mathcal{L}_A$, is thus computed as:

$$\mathcal{L}_A = \sum_{i=1}^{N}\sum_{j=1}^{N}\big[-G_{i,j}\log(\hat{G}_{i,j}) - (1-G_{i,j})\log(1-\hat{G}_{i,j})\big]. \quad (3)$$

Global Grouping Losses: Using only a local grouping loss may lead to global grouping inconsistency. However, formulating the global grouping consistency into a loss for a deep neural network is not straightforward. Our strategy is to utilise each row vector of $\hat{\mathbf{G}}$, $\hat{G}_{i,:}$, as a global grouping relationship vector to represent S_i. We then enforce constraints at two levels: (i) **Segment-level:** the segments belonging to the same group should have more similar global grouping relationships to each other, than to a segment outside the group. We implement this ternary relation in its strictest form, *i.e.*, via a hard triplet ranking loss, as follows:

$$\mathcal{L}_{\mathcal{T}} = \max(0, \Delta + d(\hat{G}_{i,:}, \hat{G}_{i^+,:}) - d(\hat{G}_{i,:}, \hat{G}_{i^-,:})), \quad (4)$$

where i represents an anchor segment. i^+, i^- are the hardest positive and negative samples mined online by calculating the most dissimilar one in the same group and the most similar one from a different group under the $d(\cdot)$ metric, respectively. Δ is a margin and $d(\cdot)$ denotes a distance function between two feature inputs. Here we take the squared Euclidean distance under the l_2 normalisation. (ii) **Group-level:** the group should be structurally compact in itself, but pushed further away with other groups. Concretely, for $\hat{G}$ with k annotated groups, we obtain each of the k centroids by averaging the global representations of the segments it belongs, denoted as:

$$\hat{G}_{centre_k} = \frac{1}{n_k}\sum_{i\in group_k}\hat{G}_{i,:} \quad (5)$$

where n_k is the amount of segments in the k-th group.

We then encourage the segments within one group on a whole to be close to its centre, while between-group centres to be well separated. This loss is essentially a variant of the supervised centre loss for classification [32] which is useful for learning deep embedding spaces where data of the same class is compact and separable from others. More specifically, the loss is formulated as:

$$\mathcal{L}_{intra-centre} = \frac{1}{k}\sum_{j=1}^{k}\frac{1}{n_j}\sum_{i\in group_j} d(\hat{G}_{i,:}, \hat{G}_{center_j})$$

$$\mathcal{L}_{inter-centre} = -\frac{1}{k(k-1)}\sum_{i=1}^{k}\sum_{j=i+1}^{k} d(\hat{G}_{center_i}, \hat{G}_{center_j}) \quad (6)$$

We combine the two losses together to form our final instance-level centre loss for grouping:

$$\mathcal{L}_{\mathcal{C}} = \mathcal{L}_{intra-centre} + \mathcal{L}_{inter-centre} \quad (7)$$

Generative Losses: For the generative branch, we use the same generative losses as in [18]. These include a reconstruction loss $\mathcal{L}_R$ and a KL loss $\mathcal{L}_{KL}$ measuring the difference between the latent random vector z and an IID Gaussian vector with zero-mean and unit variance.

Full Learning Objective: Our full loss $\mathcal{L}_F$ can be written as:

$$\mathcal{L}_F = \lambda_a \mathcal{L}_A + \lambda_t \mathcal{L}_T + \lambda_r(\mathcal{L}_R + \mathcal{L}_{KL}) + \lambda_c \mathcal{L}_C \tag{8}$$

where the hyper-parameters λ_a, λ_t, λ_r and λ_c describe the relative importance of the different losses in the full training objective.

Model Testing: During the testing stage, given a sketch, the trained model is used to compute an estimated segment affinity matrix, $\hat{G}$. This affinity matrix is then used to generate the final grouping. Since the number of groups varies for different sketches, the group number also needs to be estimated. To this end, we adopt a recently proposed agglomerative clustering method [33] to produce the final grouping. Note that the method does not introduce any additional free parameters.

D. Applications to Sketch Analysis

Sketch Semantic Segmentation: Our deep grouping model can be readily extended for sketch segmentation: Given a sketch object category, for each segment S_i, we obtain its feature representation f_i using our grouper and apply an additional fully-connected layer of softmax activation for classification of the segment into a semantic group (*e.g.*, wings of an aeroplane). Importantly, we show that when the grouper is first learned using multiple object categories and then fine-tuned for the segmentation task for each individual category, a marked performance boost is obtained – showing grouping principles learned by a universal grouper are beneficial for segmenting any object category, even when the grouper is trained from completely different categories.

Sketch Synthesis from EdgeMap: A simple sketch synthesis method can be developed based on the proposed universal grouper. The method is based on grouping edgemaps extracted from photo images and removing the least important groups. Assume that the N segments of an edgemap have been grouped in K groups, denoted as $P_k, k \in [1, K]$. An importance measure is defined as:

$$I(P_k) = I_L(P_k) \cdot I_N(P_k) + I_D(P_k) \tag{9}$$

where $I_L(P_k)$, $I_N(P_k)$ and $I_D(P_k)$ measure the importance from the perspectives of length, numbers and distribution of the segments in group P_k respectively. A less important group has smaller number of segments with shorter lengths but occupies a bigger region. We thus have:

$$\begin{aligned} I_L(P_k) &= \frac{\sum_{i=1}^{N_{P_k}} L_{S_i}}{\sum_{i=1}^{N} L_{S_i}} \quad I_N(P_k) = \frac{N_{P_k}}{N} \\ I_D(P_k) &= \frac{\max(w,h) N_{P_k}}{\sum_{i=1}^{N_{P_k}} d(M_{P_k}, M_{S_i})} \end{aligned} \tag{10}$$

where N_{P_k} is the number of segments in P_k, L_{S_i} is the length of segment S_i, w and h are the width and height of the object, respectively, M_{P_k} denotes the average position of group P_k in the image plane, M_{S_i} represents the average position of segment S_i, and Euclidean distance $d(\cdot)$ is used. With the importance measure $I(P_k)$ computed for each group, we can then drop the least important groups defined as those with $I(P_k) < I_\delta$ where I_δ is a threshold.

Fine-Grained Sketch Based Image Retrieval: We further develop an unsupervised FG-SBIR method following [19]. Specifically, we apply our grouper to edgemaps extracted from photos to synthesise human style sketches. Three threshold values of I_δ are used for each photo to accounts for the variable levels of abstraction among human sketchers. The photos and corresponding synthesised sketches are then used as data to train an off-the-shelf FG-SBIR model [21]. During testing, the grouping and group removal processes are applied to the human sketches, again with three different thresholds. The matching scores using the three abstracted sketches plus the original query sketch are then fused to produce the final retrieval results. Note that for this unsupervised FG-SBIR model to work well, our grouper must be truly universal: it needs to work well on both human sketches which it was trained on, and photo edgemaps.

V. Experiments on Perceptual Grouping

A. Datasets and Settings

Dataset Splits and Preprocessing: Among the 25 categories in the new SPG dataset, we randomly select 20 as **seen categories**, and use the remaining 5 categories as **unseen categories** to test the generalisation of our universal grouper. In each seen category, we select 650 sketches for training, 50 for validation, and 100 for testing. For the unseen categories, no data are used for training and we randomly select 100 sketches per category for testing to have the same per-category size as the seen categories. We normalise all the sketch strokes, and augment the sketch via stroke removal and distortion [27].

Implementation Details: Our deep grouper is implemented on Tensorflow on a single Titan X GPU. For model training, we set the importance weights λ_a,λ_t, λ_r and λ_c for different losses (Eq. (4)) to 0.6, 1.5, 0.5 and 0.8, respectively. The Adam optimiser [34] is applied with the parameters $\beta_1 = 0.5$, $\beta_2 = 0.9$, $\epsilon = 10^{-8}$. The initial learning rate is set to 0.0003 with exponential weight decay. The model is trained for 22,000 iterations with a batch size of 100.

Evaluation Metrics: Sketch perceptual grouping shares many common characteristics with the unsupervised image segmentation problem [6]. We thus adopt the same metrics including variation of information (VOI), probabilistic rand index (PRI), and segmentation covering (SC) as defined in [35]. More detailed definition of these metrics in the context of sketch grouping are: (i) **VOI:** the distance between two groups in terms of their average conditional entropy is calculated. (ii) **PRI:** the compatibility of assignments between pairs of stroke segments in each group is compared. (iii) **SC:** the overlapping between the machine grouping and human grouping is measured. For SC and PRI, higher scores are better, while for VOI, a lower score indicates better grouping results.

Competitors: Very few sketch perceptual grouping methods exist. The state-of-the-art model **Edge-PG** [12] uses two Gestalt principles, namely proximity (spatial closeness) and continuity (slope trend) to compute an affinity matrix and

Category	**USPG-2.0**			**USPG-1.0** [26]			**Edege-PG [12]**			**DeepLab [5]**		
	VOI↓	**PRI ↑**	**SC ↑**	**VOI↓**	**PRI ↑**	**SC ↑**	**VOI ↓**	**PRI ↑**	**SC ↑**	**VOI ↓**	**PRI ↑**	**SC ↑**
Airplane	**0.55**	**0.91**	**0.83**	0.58	0.88	0.78	0.72	0.80	0.71	1.09	0.72	0.65
Alarm clock	**0.44**	**0.94**	**0.85**	0.46	0.93	0.83	0.59	0.84	0.73	0.86	0.80	0.70
Ambulance	**0.61**	**0.90**	**0.82**	0.67	0.86	0.77	1.35	0.67	0.60	1.19	0.71	0.63
Ant	**0.74**	**0.88**	**0.79**	0.86	0.83	0.69	1.32	0.68	0.62	1.38	0.69	0.60
Apple	**0.23**	**0.93**	**0.92**	0.25	0.92	0.91	0.54	0.88	0.79	0.82	0.83	0.72
Backpack	**0.53**	**0.92**	**0.81**	0.57	0.88	0.79	1.29	0.70	0.61	1.59	0.67	0.59
Basket	**0.69**	**0.89**	**0.79**	0.76	0.84	0.74	1.27	0.71	0.59	1.37	0.69	0.61
Butterfly	**0.79**	**0.85**	**0.76**	0.83	0.76	0.65	1.30	0.69	0.58	1.58	0.66	0.58
Cactus	**0.45**	**0.92**	**0.85**	0.51	0.90	0.83	0.86	0.82	0.71	0.90	0.79	0.68
Calculator	**0.46**	**0.89**	**0.84**	0.50	0.86	0.83	0.98	0.77	0.68	1.17	0.72	0.64
Camp fire	**0.27**	**0.97**	**0.92**	0.28	0.95	0.91	1.05	0.71	0.65	0.77	0.85	0.74
Candle	**0.82**	**0.85**	**0.78**	0.89	0.78	0.69	1.47	0.65	0.57	1.54	0.67	0.60
Coffee cup	**0.35**	**0.93**	**0.88**	0.38	0.91	0.86	0.85	0.83	0.68	0.98	0.79	0.66
Crab	**0.63**	**0.89**	**0.78**	0.69	0.81	0.74	1.29	0.69	0.56	1.58	0.67	0.60
Duck	**0.79**	**0.88**	**0.77**	0.86	0.83	0.69	0.95	0.74	0.68	1.63	0.65	0.57
Face	**0.71**	**0.87**	**0.81**	0.81	0.84	0.74	1.24	0.69	0.61	0.80	0.82	0.73
Ice-cream	**0.39**	**0.95**	**0.92**	0.41	0.94	0.85	0.79	0.82	0.71	1.40	0.68	0.62
Pig	**0.55**	**0.88**	**0.83**	0.63	0.84	0.78	1.55	0.63	0.50	0.98	0.77	0.67
Pineapple	**0.44**	**0.94**	**0.88**	0.50	0.93	0.82	0.63	0.83	0.72	1.05	0.74	0.65
Suitcase	**0.48**	**0.91**	**0.89**	0.54	0.89	0.83	0.58	0.82	0.75	1.10	0.73	0.64
Average	**0.55**	**0.91**	**0.84**	0.59	0.87	0.79	1.03	0.75	0.65	1.20	0.73	0.65

TABLE I: Comparative grouping results on seen categories. Our SPG dataset.

Category	**USPG-2.0**			**USPG-1.0** [26]			**Edge-PG [12]**		
	VOI ↓	**PRI ↑**	**SC ↑**	**VOI ↓**	**PRI ↑**	**SC ↑**	**VOI ↓**	**PRI ↑**	**SC ↑**
Angel	**0.62**	**0.89**	**0.82**	0.70	0.87	0.73	1.19	0.69	0.60
Bulldozer	**0.71**	**0.89**	**0.79**	0.81	0.85	0.73	1.37	0.65	0.58
Drill	**0.60**	**0.85**	**0.82**	0.67	0.78	0.77	1.45	0.61	0.53
Flower	**0.35**	**0.92**	**0.86**	0.39	0.90	0.84	0.79	0.75	0.64
House	**0.41**	**0.92**	**0.87**	0.46	0.91	0.83	0.85	0.77	0.69
Average	**0.54**	**0.89**	**0.83**	0.64	0.86	0.77	1.13	0.69	0.61

TABLE II: Perceptual grouping results on unseen categories. Our SPG dataset.

feeds the matrix to a graph cut algorithm to get the groups. The weightings of the two principles are learned from data using RankSVM. This method thus differs from ours in that hand-crafted features are used and only two principles are modelled. Beyond sketch grouping, many semantic image segmentation methods have been proposed lately based on fully convolutional networks (FCN). We choose one of the state-of-the-art models, **DeepLab** [5] as a baseline. It is trained to take images as input and output the semantic grouping, *i.e.*, each pixel is assigned a class label. A conditional random field (CRF) is integrated to the network to enforce the proximity and similarity principles. Note that: (1) DeepLab is a supervised semantic segmentation method. It thus needs not only grouping annotation as our model does, but also group ID annotation, which is not used by our model and Edge-PG. This gives it an unfair advantage. (2) It performs grouping at the pixel level whilst both our model and Edge-FG do it at the stroke/segment level. Finally, the earlier version [26] of our full deep grouping model (**USPG**[2]**-2.0**), denoted **USPG-1.0**, was also compared. USPG-1.0 differs from the full model in that a) no hard data mining is performed for the the ranking loss and b) the center loss is not used.

B. Results

Results on Seen Categories: In this experiment, the model is trained on the seen category training set and tested on the seen category testing set. From Table I, we can see that: (i) Our model achieves the best performance across all 25 categories on each metric. The VOI improvement is particularly striking indicating that the groups discovered by our model in each sketch are distinctive to each other. In contrast, the two compared existing models tend to split a semantic part into multiple groups (see Fig. 4). (ii) Edge-PG is much worse than our method because it is based on hand-crafted features for only two principles, while our model implicitly learns the features and combination strategy based on end-to-end learning from human group annotation. (iii) Although DeepLab also employs a deep neural network and uses additional annotations, its result is no better than Edge-PG. This suggests that for sketch perceptual grouping, it is important to treat sketches as a set of strokes rather than pixels, as strokes already grouping pixels. These constraints are ignored by the DeepLab types of models designed for photographic image segmentation. (iv) Compared with its earlier version USPG-1.0, the improved USPG-2.0 is clearly superior thanks to the better formulation of the global grouping losses.

Some examples of the grouping results are shown in Fig. 4. As expected, ignoring the stroke level grouping constraint on pixels, each stroke is often split into multiple groups by DeepLab [5]. Edge-PG [12] does not suffer from that problem. However, it suffers from the limitations on modelling only two principles, *e.g.*, to group the clock contour (second column) into one group, the closure principle needs to be used. It is also unable to model even the two principles effectively due to the limited expressive power of hand-crafted features: in the airplane example (first column), the two wings should be grouped together using the continuity principle, but broken into two by Edge-PG. In contrast, our model produces more consistent groupings using multiple principles dynamically.

[2] USPG: Universal Sketch Perceptual Grouper

Method	VOI ↓	PRI ↑	SC ↑
Edge-PG [12]	1.69	0.62	0.53
USPG-1.0 [26]	0.96	0.78	0.71
USPG-2.0	**0.81**	**0.82**	**0.76**

TABLE III: Grouping comparison of Edge-PG [12], USPG-1.0 [26] with our full model USPG-2.0 on [12]'s dataset.

For instance in the cactus example (last column), to produce the correct grouping of those spikes, both continuity, similarity and less prevalent principles such as common fate need to be combined. Only our model is able to do that because it has implicitly learned to model all the principles used by humans to annotate the groupings.

Results on Unseen Categories: In this experiment, models learned using seen categories are tested directly on unseen categories without any fine-tuning. It is thus intuitively designed to evaluate whether the grouper is indeed universal, *i.e.*, can be applied to any new object category. Note that as a supervised segmentation method, DeepLab cannot be applied here because each category has a unique set of group IDs. From Table II, it can be seen that our model significantly outperforms Edge-PG and importantly, by comparing with Table I, our model's performance on PRI and SC hardly changed. In contrast, the Edge-PG's performance on the unseen categories is clearly worse than that on the seen categories. This suggests that our grouper is more generalisable and universal. Again, clear improvement over USPG-1.0 is obtained. Some qualitative results are shown in Fig. 5. We further show grouping results on more real-world object categories that are not included in SPG dataset in the supplemental material.

Results on Unseen Dataset: To further demonstrate the generalisation ability of our universal grouper, we test the trained model on a different dataset. Specifically, we choose 10 categories from the dataset in [36] including 5 categories overlapping with our dataset and 5 new categories. Note that the sketches in this datasets are from the database proposed in [37], which are drawn without the 20 second constraint, thus exhibiting much more details with better quality in general. This dataset thus represents a different domain. Table III shows that our model again demonstrates better generalisation ability.

Ablation Study: Our model is trained with a combination of generative and discriminative losses (Sec. IV-C). These include the local grouping loss $\mathcal{L}_A$, global hard triplet ranking loss $\mathcal{L}_T$, global instance-level centre loss $\mathcal{L}_C$, generative loss $\mathcal{L}_R$ and KL loss $\mathcal{L}_{KL}$. Among them, all but the KL loss can be removed, leading to several variants of our model *e.g.*, USPG-2.0−A−C−T is obtained by removing $\mathcal{L}_A$, $\mathcal{L}_C$ and $\mathcal{L}_T$. In addition, we implement USPG+l_2 which uses an l_2 to on the predicted affinity matrix $\hat{G}$ w.r.t. the ground truth matrix G, to replace all the losses formulated in our model. This is to examine the importance of having separate local and global grouping losses. The results are shown in Table IV. Clearly all four losses contribute to the performance of our model. The poorest result was obtained when an l_2 loss is used directly on the predicted affinity matrix, suggesting that balancing the local and global grouping losses is critical for learning a good grouper. We further show that the improvement of our full model over USPG-2.0−R on unseen categories (0.54 vs. 0.77) is bigger on seen categories (0.55 vs. 0.60). This indicates that the generative loss helps the model to better generalise.

VI. Experiments on Sketch Segmentation

We conduct sketch segmentation experiments on the 20 seen categories of our SPG dataset, and follow the same dataset splits and preprocessing settings as in Sec. V – 650 sketches from each of the 20 categories in the seen split for training. We also evaluate our method on the two existing sketch segmentation datasets [9], [10] using the same dataset splits and pre-processing as in [10], and adopt their evaluation metrics throughout this section: (i) **P-metric**: pixel/segment-level agreement between the prediction and ground-truth labels; (ii) **C-metric**: the component/part prediction accuracy. We assume a component/part is correctly labelled if 75% of its pixels/segments are assigned the correct label.

Competitors: Few sketch segmentation methods exist, again due to lack of large scale datasets. We compare with the state-of-the-art sketch segmentation model **Sketch-CRF** [10], which combines the hand-crafted feature with CRF to assign each segment a semantic label, and popular image segmentation model **DeepLab** [5], which is pre-trained on ImageNet and PASCAL and adapted by fine-tuning the fully-connected layers only and keeping the remaining convolutional layer weights fixed. Since human sketch presents very sparse visual cues with the majority of pixels acting as background, we ignore non-sketch pixels during training. As mentioned earlier, our grouper is repurposed for segmentation by adding an additional classification layer on top of the feature representation for each segment. we consider two variants **SEG+PG** and **SEG**, depending on whether the perceptual grouping (PG) losses introduced in this work are still employed together with the group ID prediction loss. Finally, we consider another model termed **SEG+PG+Pre-train** which pretrains the segmentation model on the grouping task over multiple categories before fine-tuning on individual categories.

Implementation Details: We adopt the same set of weighting hyper-parameters for each component as in perceptual grouping experiment, with the additional weight for ID prediction loss set as 1. For experiments that receive raster image input (*e.g.*, DeepLab), we normalise the maximum height and width of each vector object to 200 pixels and put it on the centre of a 256 × 256 blank canvas before converting to the PNG format. Note that for the segmentation experiment, we need to train one model per category, since each category may contain different number of semantic parts.

Results on Sketch Segmentation: From Table V, VI, VII, the following observations can be made: (i) Compared with the existing baselines, SEG consistently performs better across all metrics and datasets on most categories, suggesting that the choice of temporal modelling of the sketches as sequences of strokes and deep data-driven feature learning help. (ii) Integrating the proposed perceptual grouping losses into the sketch segmentation model training is beneficial, as indicated

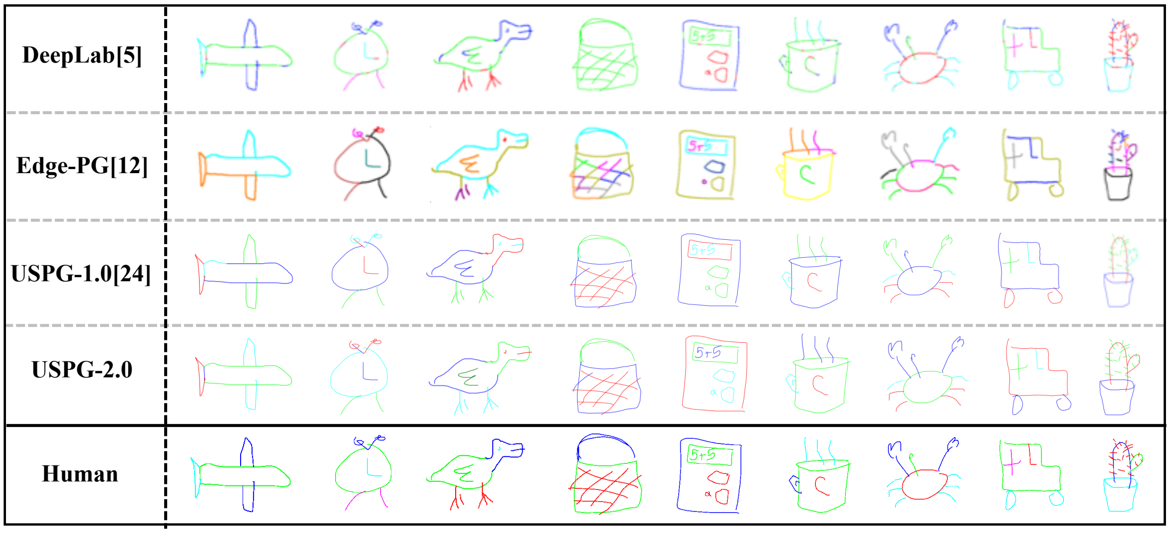


Fig. 4: Qualitative grouping results on seen categories.

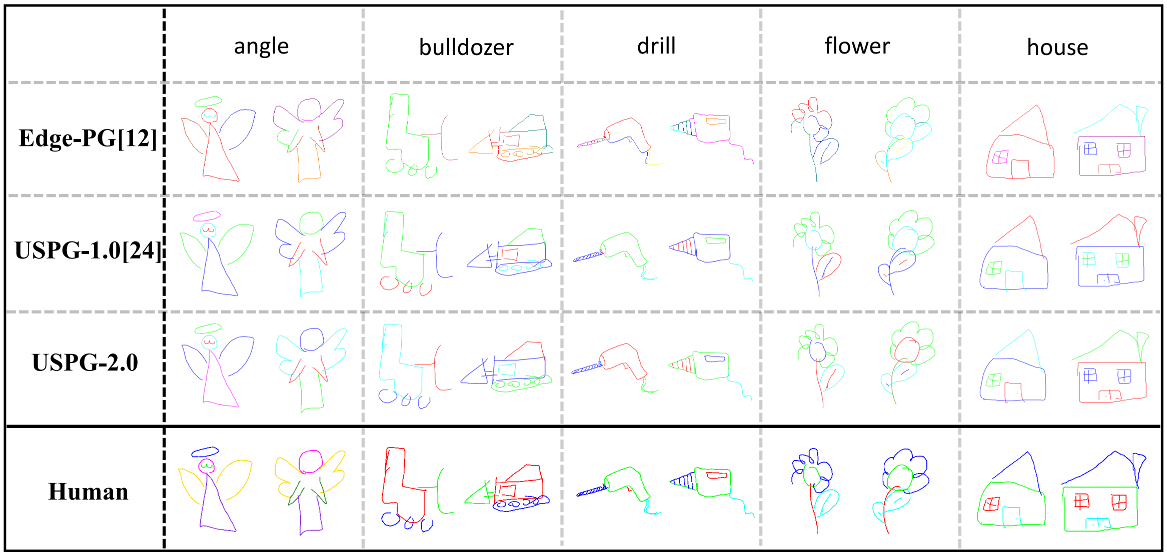


Fig. 5: Qualitative grouping results on unseen categories.

by the better performance of SEG+PG over SEG. (iii) The usefulness of SPG as a pre-training step for the model learned on other sketch segmentation datasets is also observed: both Tables VI and VII show that the best overall performance is achieved by SEG+PG+Pre-train. This suggests that some general perceptual principles learned across categories can be generally applicable.

VII. Experiments on Sketch Synthesis and FG-SBIR

One application of our grouper is to use it as an abstraction model so that edgemaps extracted from photos can be grouped and abstracted to synthesise human-like sketches. These pseudo sketches are then used to train a state-of-the-art FG-SBIR model [21] without using any real human sketches. We conduct experiment on the largest FG-SBIR datasets QMUL Shoe-V2 and Chair-V2 [39].

Competitors: We first compare with the same FG-SBIR model trained using synthesised sketches from the deep conditional GAN network in [38] (denoted as **Scribbler**). It is a standard encoder-decoder image-to-image translation model with residual convolutional blocks. We further compare with a recently proposed unsupervised FG-SBIR model **LDSA** [19] which is also based on abstracting photo edgemaps to synthesise sketches via deep reinforcement learning. We also illustrate the **Upper-Bound** which is obtained using the same FG-SBIR model trained with the real sketch-photo pairs in Shoe-V2 and Chair-V2.

Implementation Details: We use off-the-shelf post-processing toolkit (*e.g.*, AutoTrace) to transform a raster edgemap image into a time sequence input before feeding into our deep perceptual grouping model, following similar practice in [19]. For all FG-SBIR competitors, we only tuned the margin (denotes as Δ in Eq.(2) in [21]) to 0.8.

Method	Seen Categories			Unseen Categories		
	VOI ↓	PRI ↑	SC ↑	VOI ↓	PRI ↑	SC ↑
USPG-2.0−A−C−T	1.45	0.65	0.59	1.53	0.64	0.56
USPG-2.0−R−C−T	1.12	0.71	0.64	1.36	0.68	0.59
USPG-2.0−A−R−C	1.27	0.69	0.63	1.48	0.64	0.57
USPG-2.0−A−R−T	1.08	0.69	0.65	1.29	0.69	0.60
USPG-2.0−C−T	0.63	0.86	0.78	0.71	0.84	0.73
USPG-2.0−C−R	0.63	0.85	0.77	0.81	0.80	0.70
USPG-2.0−T−R	0.65	0.84	0.76	0.83	0.79	0.69
USPG-2.0−T	0.57	0.88	0.81	0.58	0.87	0.79
USPG-2.0−C	0.56	0.89	0.82	0.57	0.87	0.80
USPG-2.0−R	0.60	0.87	0.79	0.77	0.81	0.71
USPG+l_2	2.11	0.64	0.55	2.32	0.61	0.53
USPG-1.0 [26]	0.59	0.87	0.79	0.64	0.86	0.77
USPG-2.0	**0.55**	**0.91**	**0.84**	**0.54**	**0.89**	**0.83**

TABLE IV: Grouping performance of different variants of our model on seen and unseen categories.

Category	SEG		SEG+PG		DeepLab [5]	
	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric
Airplane	81.5%	70.4%	**82.9%**	**70.9%**	70.7%	46.2%
Alarm clock	83.5%	80.2%	**84.8%**	**81.0%**	82.5%	74.3%
Ambulance	77.3%	62.7%	**80.7%**	**68.1%**	72.5%	54.2%
Ant	59.3%	47.3%	**66.4%**	**56.6%**	61.3%	32.1%
Apple	89.8%	71.2%	**89.9%**	**71.8%**	87.3%	60.2%
Backpack	71.0%	56.0%	**75.2%**	**63.7%**	64.3%	28.4%
Basket	84.3%	79.8%	**84.8%**	**83.2%**	79.5%	69.5%
Butterfly	88.6%	81.3%	**89.0%**	**83.6%**	85.6%	69.8%
Cactus	74.8%	69.0%	**77.5%**	**72.3%**	67.2%	30.8%
Calculator	89.2%	85.2%	91.1%	89.9%	**92.5%**	**92.1%**
Camp fire	91.5%	90.8%	**92.3%**	**91.4%**	82.9%	83.3%
Candle	85.6%	69.3%	88.3%	71.8%	**91.5%**	**76.9%**
Coffee cup	90.2%	85.3%	**92.0%**	**87.2%**	86.2%	81.8%
Crab	73.5%	63.8%	**77.9%**	**70.5%**	73.9%	49.3%
Duck	82.1%	70.5%	**86.9%**	**75.4%**	85.9%	76.0%
Face	85.9%	78.1%	**88.0%**	**80.1%**	87.4%	78.4%
Ice-cream	83.6%	76.5%	**85.4%**	**79.3%**	80.7%	70.3%
Pig	78.7%	71.4%	81.9%	75.4%	**82.1%**	**77.9%**
Pineapple	88.6%	90.1%	**89.8%**	**90.2%**	85.4%	79.5%
Suitcase	91.6%	89.2%	**92.7%**	**90.7%**	90.2%	90.1%
Average	82.5%	74.4%	**84.9%**	**77.6%**	80.5%	59.0%

TABLE V: Comparative sketch segmentation results on our SPG dataset.

Category	Sketch-CRF [10]		SEG		SEG+PG		SEG+PG+Pre-train		DeepLab [5]	
	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric
Airplane	55.1%	48.7%	69.5%	54.8%	75.2%	61.0%	**78.5%**	**64.9%**	65.1%	45.7%
Bicycle	79.7%	68.6%	80.7%	74.7%	85.0%	81.3%	**86.4%**	**83.8%**	78.2%	65.9%
Cdlbrm	72.0%	66.2%	77.3%	70.8%	**88.4%**	**82.5%**	86.7%	81.2%	75.8%	66.8%
Chair	66.5%	61.6%	76.0%	71.9%	87.8%	**88.0%**	**88.5%**	87.3%	73.6%	66.2%
Fourleg	81.5%	74.2%	84.1%	75.8%	**90.3%**	**81.6%**	88.2%	80.6%	85.3%	75.1%
Human	69.7%	63.1%	73.2%	65.8%	72.8%	65.3%	**75.7%**	**68.4%**	69.6%	60.9%
Lamp	82.9%	77.2%	88.3%	82.9%	**95.3%**	**92.7%**	93.1%	90.6%	84.2%	77.8%
Rifle	67.8%	65.1%	71.2%	73.2%	70.9%	72.0%	**75.4%**	**74.9%**	68.3%	64.9%
Table	74.5%	65.6%	79.2%	70.9%	84.1%	80.6%	**85.8%**	**81.3%**	77.0%	66.4%
Vase	83.3%	79.1%	85.2%	80.4%	**89.5%**	**82.8%**	87.5%	80.6%	84.5%	77.5%
Average	73.2%	67.0%	78.5%	72.1%	83.9%	78.8%	**84.6%**	**79.4%**	76.2%	66.7%

TABLE VI: Comparative sketch segmentation results on [9] dataset.

Results on Sketch Synthesis: Fig. 6 shows some qualitative examples of edgemap grouping results and the synthesised sketches. It can be seen that our grouper is generalisable to photo edges and our abstraction method produces visually

Category	Sketch-CRF [10]		SEG		SEG+PG		SEG+PG+Pre-train		DeepLab [5]	
	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric	P-metric	C-metric
Airplane	74.6%	76.2%	76.8%	76.9%	81.7%	77.8%	**83.2%**	**79.8%**	76.1%	70.8%
Butterfly	77.7%	78.0%	79.3%	79.9%	83.6%	80.3%	**84.5%**	**81.3%**	78.9%	73.6%
Face	88.9%	86.0%	90.8%	87.6%	**93.1%**	**89.6%**	91.8%	88.2%	89.4%	83.5%
Flower with Stem	74.5%	73.0%	83.9%	81.8%	**88.4%**	**85.2%**	87.6%	83.7%	81.1%	76.4%
Pineapple	**96.9%**	**96.2%**	94.7%	90.6%	96.2%	95.7%	95.8%	94.1%	92.4%	89.9%
Snowman	85.2%	81.7%	86.5%	79.8%	89.5%	82.4%	**91.3%**	**87.8%**	87.1%	78.7%
Average	83.0%	81.8%	85.3%	82.8%	88.8%	85.2%	**89.0%**	**85.8%**	84.2%	78.8%

TABLE VII: Comparative sketch segmentation results on the dataset in [10] .

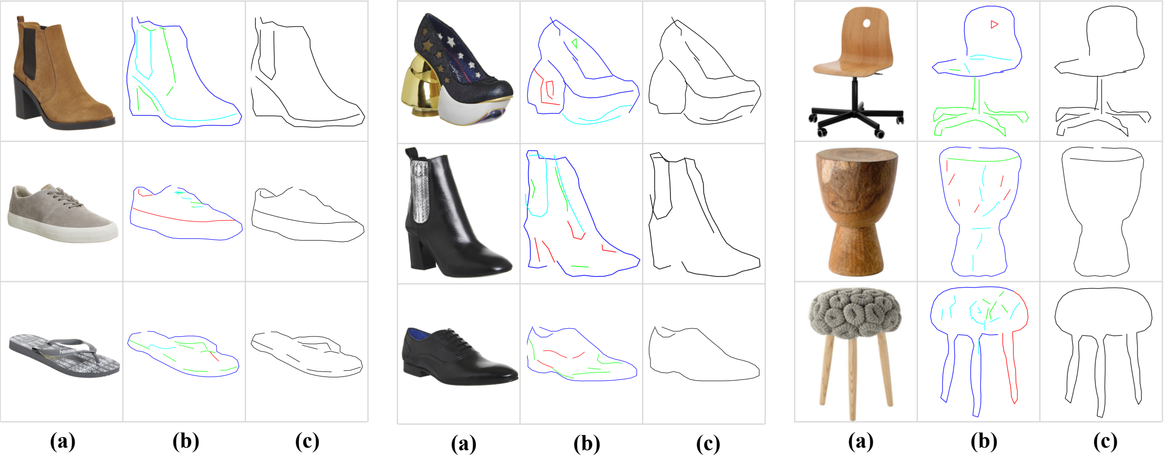


Fig. 6: Applying our grouper to synthesise abstract sketches from photo edgemaps. **(a)** columns show the photos; **(b)** columns give the edgemaps extracted from the photos and the grouping results; **(c)** columns provide synthesised abstract sketches.

Method	Shoe-V2		Chair-V2	
	Top1	Top10	Top1	Top10
Scribbler [38]	8.86%	32.28%	31.27%	78.02%
LDSA [19]	21.17%	55.86%	41.80%	84.21%
USPG-2.0	26.88%	61.86%	45.57%	88.61%
Upper Bound	34.38%	79.43%	48.92%	90.71%

TABLE VIII: FG-SBIR performance on Shoe-V2 and Chair-V2 datasets.

appealing sketches. Note that we do not seek to synthesise human-like sketches as in [14], [15], which is itself an open problem that requires careful stroke-level treatment and/or explicit temporal modelling of the sketching process. The sole purpose of sketch synthesis in the context of this work is to demonstrate the universal applicability of our grouper, that it not only works on sketch data, but on edgemaps as well being a modality that has not been observed during training

Results on FG-SBIR: As can be seen in Table VIII: (i) our model performs much better than Scribbler. This suggests that our edge abstraction model, albeit simple, synthesises more realistic sketches from edgemaps. (ii) Our model outperforms LDSA model by 5.71% and 3.63% on top 1 accuracy on Shoe-V2 and Chair-V2, respectively. (iii) Our results are not far off the Upper-Bound.This shows that our method enables FG-SBIR to be used without the expensive collection of sketch-photo pairs.

VIII. Conclusion

We have proposed an end-to-end sketch perceptual grouping model. This is made possible by collecting a new large-scale sketch grouping dataset SPG. Our grouper is trained with generative losses to make it generalisable to new object categories and datasets/domains. A number of grouping losses were also formulated to balance the local and global grouping constraints. Extensive experiments showed that our model significantly outperforms existing groupers. We also demonstrated our grouper's application to sketch segmentation, sketch synthesis and FG-SBIR. Ongoing work includes the investigation on how to use the learned perceptual grouping principles to other grouping/segmentation tasks such as semantic image segmentation.

References

[1] J. Wagemans, J. H. Elder, M. Kubovy, S. E. Palmer, M. A. Peterson, M. Singh, and R. von der Heydt, "A century of gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization." *Psychological bulletin*, 2012.

[2] J. Wagemans, J. Feldman, S. Gepshtein, R. Kimchi, J. R. Pomerantz, P. A. van der Helm, and C. van Leeuwen, "A century of gestalt psychology in visual perception: II. Conceptual and theoretical foundations." *Psychological bulletin*, 2012.

[3] M. Wertheimer, *Laws of organization in perceptual forms.* London, England: Kegan Paul, Trench, Trubner & Company, 1938.

[4] X. Ren and J. Malik, "Learning a classification model for segmentation," in *ICCV*, 2003.

[5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *arXiv:1606.00915*, 2016.

[6] X. Xia and B. Kulis, "W-net: A deep model for fully unsupervised image segmentation," *ArXiv e-prints*, 2017.

[7] C. Wang, B. Yang, and Y. Liao, "Unsupervised image segmentation using convolutional autoencoder with total variation regularization as preprocessing," in *ICASSP*, 2017.

[8] Z. Sun, C. Wang, L. Zhang, and L. Zhang, "Free hand-drawn sketch segmentation," in *ECCV*, 2012.

[9] Z. Huang, H. Fu, and R. W. Lau, "Data-driven segmentation and labeling of freehand sketches," *TOG*, 2014.

[10] R. G. Schneider and T. Tuytelaars, "Example-based sketch segmentation and labeling using crfs," *TOG*, 2016.

[11] Y. Qi, J. Guo, Y. Li, H. Zhang, T. Xiang, and Y.-Z. Song, "Sketching by perceptual grouping," in *ICIP*, 2013.

[12] Y. Qi, Y. Z. Song, T. Xiang, H. Zhang, T. Hospedales, Y. Li, and J. Guo, "Making better use of edges via perceptual grouping," in *CVPR*, 2015.

[13] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 2006.

[14] J. Song, K. Pang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Learning to sketch with shortcut cycle consistency," in *CVPR*, 2018.

[15] Y. Li, Y. Z. Song, T. M. Hospedales, and S. Gong, "Free-hand sketch synthesis with deformable stroke models," *IJCV*, 2015.

[16] A. Amir and M. Lindenbaum, "A generic grouping algorithm and its quantitative analysis," *TPAMI*, 1998.

[17] Z. Lun, C. Zou, H. Huang, E. Kalogerakis, P. Tan, M.-P. Cani, and H. Zhang, "Learning to group discrete graphical patterns," *TOG*, 2017.

[18] D. Ha and D. Eck, "A neural representation of sketch drawings," *arXiv preprint arXiv:1704.03477*, 2017.

[19] U. R. Muhammad, Y.-Z. Song, T. Xiang, and T. Hospedales, "Learning deep sketch abstraction," in *CVPR*, 2018.

[20] Y. Li, T. M. Hospedales, Y.-Z. Song, and S. Gong, "Fine-grained sketch-based image retrieval by matching deformable part models," in *BMVC*, 2014.

[21] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy, "Sketch me that shoe," in *CVPR*, 2016.

[22] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, "The sketchy database: Learning to retrieve badly drawn bunnies," in *SIGGRAPH*, 2016.

[23] J. Song, Y.-Z. Song, T. Xiang, T. Hospedales, and X. Ruan, "Deep multi-task attribute-driven ranking for fine-grained sketch-based image retrieval," in *BMVC*, 2016.

[24] K. Li, K. Pang, Y.-Z. Song, T. M. Hospedales, T. Xiang, and H. Zhang, "Synergistic instance-level subspace alignment for fine-grained sketch-based image retrieval," *TIP*, 2017.

[25] K. Pang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Cross-domain generative learning for fine-grained sketch-based image retrieval," in *BMVC*, 2017.

[26] L. Ke, K. Pang, J. Song, Y.-Z. Song, T. Xiang, T. M. Hospedales, and Z. Honggang, "Universal sketch perceptual grouping," in *ECCV*, 2018.

[27] Q. Yu, Y. Yang, F. Liu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net: A deep neural network that beats humans," *IJCV*, 2017.

[28] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," *Computer Science*, 2015.

[29] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *ArXiv e-prints*, 2013.

[30] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[31] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *TSP*, 1997.

[32] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *ECCV*, 2016.

[33] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *CVPR*, 2016.

[34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[35] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, 2011.

[36] Y. Qi, J. Guo, Y. Z. Song, T. Xiang, H. Zhang, and Z. H. Tan, "Im2sketch: Sketch generation by unconflicted perceptual grouping," *Neurocomputing*, 2015.

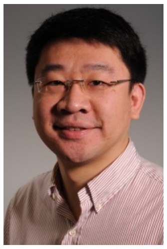
[37] M. Eitz, J. Hays, and M. Alexa, "How do humans sketch objects?" *TOG*, 2012.

[38] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays, "Scribbler: Controlling deep image synthesis with sketch and color," *CVPR*, 2017.

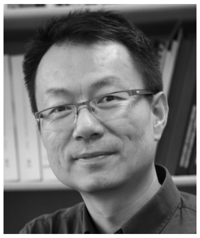
[39] Q. Yu, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "SketchX – Shoe/Chair fine-grained SBIR dataset," http://sketchx.eecs.qmul.ac.uk, 2017.

Ke Li is currently a Ph.D candidate at School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. His research interest is computer vision, particularly focus on sketch.

Kaiyue Pang is currently a Ph.D candidate at SketchX Research Lab, Queen Mary University of London. His research interest is computer vision, particularly focus on generaitve and discriminative modelling of human sketches and how such can be transferred into novel commercial applications.



Yi-Zhe Song is a Senior Lecturer (Associate Professor), and the founding Director of SketchX Research Lab, in the School of Electronic Engineering and Computer Science, Queen Mary University of London. He is interested in all problems associated with understanding human sketches, and how such understanding can be transferred into commercial applications. Prior to Queen Mary, he has worked as Research and Teaching Fellow at University of Bath, where he researched into problems such as perceptual grouping, image segmentation, cross-domain image analysis and non-photorealistic rendering.



Tao Xiang received the BS degree from Xi'an Jiaotong University, Xi'an, China, in 1995, and the PhD degree from the National University of Singapore, in 2001. He is a professor of computer vision and multimedia in the School of Electronic Engineering and Computer Science, Queen Mary University of London. His research interests include computer vision, pattern recognition and machine learning.

Timothy M. Hospedales is a Reader within IPAB in the School of Informatics at the University of Edinburgh, and Visiting Reader at Queen Mary University of London. His research focuses on machine learning, particularly life-long transfer and active learning, with both probabilistic and deep learning approaches. He has looked at a variety application areas including computer vision (behaviour understanding, person re-identification, attribute and zero-shot learning), robotics, sensor fusion, novel human-computer interfaces, computational social sciences, theoretical neuroscience and business data analytics.

Honggang Zhang received the B.S degree from the department of Electrical Engineering, Shandong University in 1996, the Master and Ph.D degrees from the School of Information Engineering, Beijing University of Posts and Telecommunications (BUPT) in 1999 and 2003 respectively. He worked as a Visiting Scholar in School of Computer Science, Carnegie Mellon University (CMU) from 2007 to 2008. He is currently an Associate Professor and Director of web search center at BUPT. His research interests include image retrieval, computer vision and pattern recognition. He published more than 30 papers on TPAMI, SCIENCE, Machine Vision and Applications, AAAI, ICPR, ICIP. He is a senior member of IEEE.