Online Sketch Recognition with Incremental Fuzzy Models

Lukas Tencer, Marta Režnáková, Mohamed Cheriet

Online Sketch Recognition with Incremental Fuzzy Models

Lukas TENCER [a], Marta REŽNÁKOVÁ [a] and Mohamed CHERIET [a]
[a] *École de technologie supérieure (Département de Génie de la Production Automatisée)*
1100 Rue Notre-Dame Ouest
H3C 1K3, Montreal, CANADA

Abstract. In this paper, we present a novel method for recognition of handwritten sketches. Unlike previous approaches, we focus on online retrieval and ability to build our model incrementally, thus we do not need to know all the data in advance and we can achieve very good recognition results after as few as 15 samples. The method is composed of two main parts: feature representation and learning and recognition. In feature representation part, we utilize SIFT-like feature descriptors in combination with soft response Bag-of-Words techniques. Descriptors are extracted locally using our novel sketch-specific sampling strategy and for support regions we follow patch-based approach. For learning and recognition, we use a novel technique based on fuzzy-neural networks, which has shown good performance in incremental learning. The experiments on state-of-the-art benchmarks have shown promising results.

1. Introduction

"It is better to see once than to hear a hundred times." Is saying old Russian proverb, which clearly favorizes visual form of communication. Since ancient times, it was specifically sketch, which allowed people to communicate visual information, record memories. Even after thousands of years, sketching is one of few ways, how majority of people can render their mental images (see Fig. 1). Since direct visualization of mental images using "mind-reading" techniques is clearly progressing (Miyawaki et al., 2008). Though they are far away from practical use and therefore sketch is the momentarily the best option for a human being to capture mental image. Also, as proved by recent research results (Walther, Chai, Caddigan, Beck, & Fei-Fei, 2011), sketches are sufficient enough to create stimuli at the same level as real-world images. This fact also justifies our choice of features based on edges.

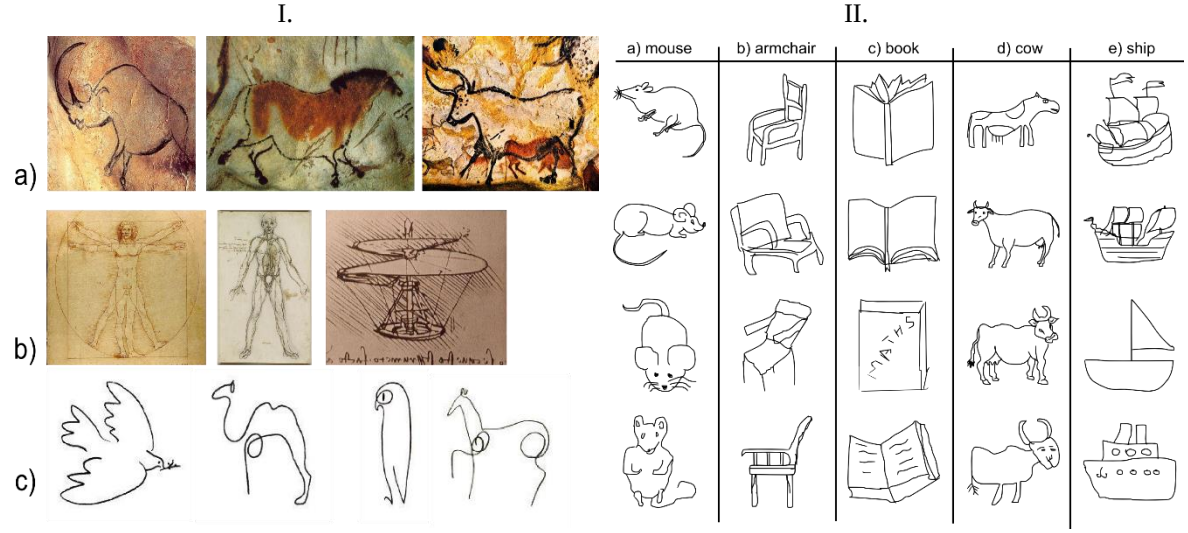

Figure 1. I. Sketches through the history. a) cave painting b) sketches from Leonardo da Vinci c) sketches from Pablo Picasso; II. Examples from our database

Unlike batch approaches, our approach does not rely on the fact, that the whole dataset is available a priori. In real-world, applications should be adaptable to the user, and it should be intelligent to learn new examples. To have one model, which represents the system "forever" is not feasible. As the user is using the sketching system more and more, his performance improves and the old model becomes obsolete and cannot capture user's performance anymore, as noted before for gestures. Therefore, we aim to develop a model, which can learn incrementally and adapt to user's drawing performance. At last, even though prior works did have available wide variety of data, this statement is not always true for commercial applications. Once sketch-based interaction is integrated into the system, it needs to capture user's sketching performance as soon as possible, with minimum number of examples. Therefore, our second motivation in the learning part of our work is to learn from scratch, with a minimum amount of prior learning data.

Prior works in the area of sketch-based focused mostly on area-specific recognition of sketches within very limited domain. These include user interfaces, chemical diagrams, architectural designs, faces or mathematical

equations (Caetano, Goulart, Fonseca, & Jorge, 2002) (Ouyang & Davis, 2007) (Tang & Wang, 2004) (Jr & Zeleznik, 2007). Therefore presence of structure and prior knowledge about the domain allows high recognition rates in these cases. Other prior works focused on more general applications and they approached sketch recognition independent of the domain of application. These techniques, are usually based on graphical models (T. Sezgin & Davis, 2007) (T. M. Sezgin & Davis, 2008) (T. M. Sezgin & Davis, 2005) (Alvarado & Davis, 2004) and require significant amount of data during the training phase, or they generate training data artificially by applying noise function to examples (T. M. Sezgin & Davis, 2005). Since we are aiming incremental learning, none of these approaches satisfies our primal condition on a good performance with a low amount of data. The most advanced approach so-far was introduced by Eitz (Eitz, Hays, & Alexa, 2012), although this one processed images in batch-manner; thus no incremental learning was performed.

2. Sketch Representation

As an input for our recognition system at learning and recognition stages is a binary image, which represents sketched image. In our method we focus on descriptors, which emphasize information abundant in sketches, that is edge orientation. After experimentation with various descriptors (Histogram Of Gradients (HOG), Edge Histogram Descriptor (EHD)), we observed best results for Scale Invariant Feature Transform (SIFT) descriptor and therefore we decided to use it in our method. Although original SIFT method as presented by (Lowe, 2004) is composed of two separate parts, interest point detection and feature extraction, we need to adapt the original technique for application to sketched images.

Our proposed descriptor is patch-based and calculated on local support region. Since many previous applications used densely sampled small regions, in case of sketches we need to lean toward larger support regions, because small regions do not capture sufficient amount of sketched regions. Size of our region depends on size of the image and amount of sampled points, so that area covered by sampled patches is about $Q = 50\ times$ the size of the image ($p\_width, p\_height = \sqrt{(img\_size^2 \cdot Q)/n\_points}$). We have experimented with different amounts of sampled area and we have found minimal gain in increase over 50, although an increase in computational cost was significant. This yields for images of size 256×256 as they are stored in testing dataset, size of support region that accounts for 9% of total image area for each patch, which is at size of about 70 pixels with 600 sampled interest points.

Our main adaptation for SIFT-like descriptors for sketches lies in a change of interest point detector. As noted by (Eitz, Hildebrand, Boubekeur, & Alexa, 2010), most prominent interest points lies on sketched lines. We would like to add an assumption, that it is also in between and in close distance to sketched lines, where we can find fine regions to sample interest points. Therefore we propose interest point detector, based on importance sampling on, between and around sketched lines, with a soft gradient between importance of different regions.

Once we have obtained sampled interest points, we calculate a descriptor on a given support region. We subdivide the image into 4×4 grid and calculate orientation histogram for each of the regions. Final descriptor is created by concatenation of the histograms for all of the regions, contribution of each pixel to the histogram of the region is weighted by Gaussian placed in the middle of the region. Normalization is applied to achieve better scale invariance. The final representation of the image is then collection of features $F = \{f_i\}$, where f_i is descriptor extracted for single local patch.

The final descriptor for representing sketches is calculated based on bag-of-words representation. For this representation, we first need to acquire a visual codebook, which we will use to encode the sketch. We construct the visual codebook by clustering the space of descriptors into k disjunct clusters, so the inner cluster scatter is minimal. The vocabulary of visual words is then represented as $V = \{v_i\}$.

Once we obtain V, then we can represent the image as a frequency histogram of visual words, where each extracted descriptor is assigned to the nearest bin given L_2 distance. Although this can be further improved by considering "soft" response histogram. In this version of the descriptor, not frequency is stored, but the relative distance to each of the visual words, this is accumulated for all the extracted local patches. Gaussian kernel is used to determine the distance between visual word and a given sample.

3. Sketch Recognition

For the recognition part, we use an online learning model, where the samples are learned incrementally and inference is calculated in real time. Thus, we will divide this section into learning and inference and describe the method used for this work. The whole model described in this paper is a hybrid ART (Adaptive Resonance Theory) and TS (Takagi-Sugeno) fuzzy neural networks originally created for online handwritten recognition.

Learning of the model is composed of two parts: generating rules for TS network and learning parameters of the rules. Generation of the rules is thus driven by ART-2A neural network, which is self-adaptive unsupervised clustering method. Here, the number of rules is not necessary set and does not equal the number of classes in the system. This is following the fuzzy logic, where all classes are defined by the possibility of occurrence within each rule.

The learning process of rule manipulation is based on an update of committed rule in a case of resonance and generating of a new rule in a case of reset. To decide this, a choice function (1) is compared with a vigilance ρ. If (2) is satisfied, resonance occurs, otherwise the reset is detected.

$$t_j = \begin{cases} x \cdot w_j & if\ j\ is\ index\ of\ committed\ node \\ \alpha \cdot \sum_{k=1}^{d} i_k & if\ j\ is\ index\ of\ uncommitted\ node \end{cases} \quad (1)$$

$$T_J = \max_j t_j \geq \rho \quad (2)$$

Then, the rule to be updated is either winning one (if resonance) or a new rule (if reset) and the update is performed (3), where w_j is a weight vector and λ a learning parameter.

$$w_j^{new} = \mathcal{N}(\lambda \cdot x + (1-\lambda) \cdot w_j^{old}) \quad (3)$$

After clustering updates, the TS network is to be learned. Each rule is in a form of (4), where both IF and THEN (antecedent and consequent) parts are learned separately. For antecedent part, we are using the incremental density update (5-7).

$$R_i: IF\ x\ is\ P_i\ THEN\ y_i^1 = \pi_i^1 x, \dots, y_i^c = \pi_i^c x \quad (4)$$

$$\beta_i = \frac{N_i}{N_i + N_i\alpha_i - 2x\rho_i + \gamma_i} \quad (5)$$

$$N_i = N_{i,old} + 1 \qquad \alpha_i = x^2 \quad (6)$$

$$\gamma_i = \gamma_{i,old} + x_{old}^2 = \gamma_{i,old} + \alpha_{i,old}, \quad \gamma_{i,init} = 0 \qquad \rho_i = \rho_{i,old} + x_{old}, \quad \rho_{i,init} = 0 \quad (7)$$

When learning the THEN part, the learning is not competitive as in the previous parts, but based on fuzzy logic. Thus, for each sample all the rules are updated with a proper increment (8-9).

$$\Pi_i = \Pi_{i,old} + C_i\beta_i x\left(y - \Pi_{i,old}\beta_i x\right); \quad \Pi_i = \{\pi_i^j\} \quad (8)$$

$$C_i = C_{i,old} - \frac{C_{i,old}\beta_i x \beta_i x^T C_{i,old}}{1 + \beta_i x^T C_{i,old}\beta_i x} \quad (9)$$

The inference if based purely on TS fuzzy network, where as shown in (4), the fuzzy results of each rule y_i^j are calculated as linear combinations of proper parameters and input sample. Then, their results are weighted by the antecedent density (5) and a final inference for every class in the system is derived (10). Then the choice of the winning class is set by the maximum over all such inferences.

$$y^j = \sum_{i=1}^{r} \beta_i \pi_i^j x \quad (10)$$

5. Experiments and Results

In this work, we have used state-of-the-art dataset of sketched images collected by Eitz (Eitz et al., 2012). It consists of 20,000 sketches in 250 categories (see Fig. 1). Categories consists of objects regularly encountered in everyday life and are aimed to capture general semantics of objects. Best reported results on this dataset are by (Eitz et al., 2012) at 56% (where chance is 4%), although it processes the data in batch manner, not in incremental manner.

We evaluate the performance of our system during the whole process of learning, thus precision should be high with every incoming sample. As evaluation criterion, we use several metrics. First criterion is simple accuracy evaluated as a ratio correctly classified element over total number of processed elements until current time t_i. In second criterion, we change the success recognition criterion. We consider an example to be recognized correctly, if correct label is one of first n returned examples, where n is set to be 2% of total number of classes. At last, we use fall-off function to increase the effect of recent errors and decrease penalization for errors, which happened in a distant past. We use two falloff functions, linear and Gaussian with a cut-off threshold at 95% of values.

As we can see Fig. 2, our results are very promising, although at the beginning of the training the performance is low. This is mostly due to low number of examples present for a given class. Also according to our observations, errors are more frequent, when new class is introduced. Using evaluation criterion of top n samples, we can see increased accuracy, even at the beginning of training. One can observe qualitative results in Fig. 2. where we present top n labels for selected queries. At last, we can see the recognition rate for the whole learning process in Fig. 2. Decreased performance in the beginning is caused by insufficient number of labeled samples.

Our method is capable of performance in real-time and execution of incremental learning and recognition of a single example takes about 340ms on standard desktop PC.

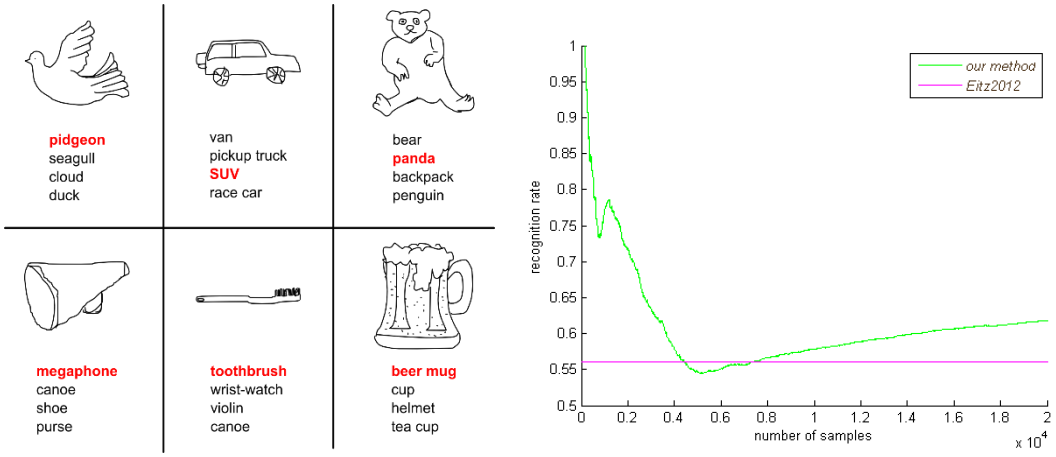

Figure 2. a) Results of the recognition algorithm, showing top 4 results. True label is highlighted in red. b) And Recognition rate evaluation, comparison of our method and the best result of state-of-the-art Eitz2012 method

6. Conclusion and Future Work

We presented a method capable of retrieval of sketched objects of everyday life. This method can process incoming data in incremental manner and is capable of learning the representation of classes in interactive and on-line mode. This is achieved through a combination of special sketch-specific features and incremental fuzzy-neural learning method. Up to our knowledge, there is no state-of-the-art method capable of incremental learning of sketched images, which can over-perform our technique.

Although our system is working in incremental manner, it still needs pre-processing to obtain the codebook. To remove this obstacle we need to devise an efficient unsupervised incremental learning algorithm, so besides incremental learning in feature space, we can also construct incrementally the visual codebook. Also the representation of the codebook itself is shallow, and we may consider higher level hierarchy to represent composite primitives as hierarchy levels in the codebook, so we can achieve higher rate of recognition. At last we are looking into combining visual and semantic retrieval for sketch-based image recognition, thus we will develop an approach to combine these two slightly distant metaphors.

The authors would like to thank to SSHRC Canada and NSERC Canada for their financial support.

References

Alvarado, C., & Davis, R. (2004). SketchREAD: a multi-domain sketch recognition engine. *Proceedings of the 17th Annual ACM Symposium …*, *6*(2).

Caetano, A., Goulart, N., Fonseca, M., & Jorge, J. (2002). Javasketchit: Issues in sketching the look of user interfaces. *AAAI Spring Symposium*

Eitz, M., Hays, J., & Alexa, M. (2012). How do humans sketch objects? *ACM Transactions on Graphics (TOG)*, *31*(4), 1–10.

Eitz, M., Hildebrand, K., Boubekeur, T., & Alexa, M. (2010). Sketch-Based Image Retrieval: Benchmark and Bag-of-Features Descriptors. *Visualization and Computer Graphics, IEEE Transactions on*, *17*(11), 1624–1636.

Jr, J. L., & Zeleznik, R. (2007). MathPad 2: a system for the creation and exploration of mathematical sketches. *ACM SIGGRAPH 2007 Courses*.

Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Miyawaki, Y., Uchida, H., Yamashita, O., Sato, M., Morito, Y., Tanabe, H. C., … Kamitani, Y. (2008). Visual image reconstruction from human brain activity using a combination of multiscale local image decoders. *Neuron*, *60*(5),

Ouyang, T., & Davis, R. (2007). Recognition of hand drawn chemical diagrams. *Proceedings of the National Conference on Artificial …*, 846–851.

Sezgin, T., & Davis, R. (2007). Sketch interpretation using multiscale models of temporal patterns. *Computer Graphics and Applications, …*, (February), 28–37.

Sezgin, T. M., & Davis, R. (2005). HMM-based efficient sketch recognition. In *Proceedings of the 10th international conference on Intelligent user interfaces* (pp. 281–283). ACM.

Sezgin, T. M., & Davis, R. (2008). Sketch recognition in interspersed drawings using time-based graphical models. *Computers & Graphics*, *32*(5), 500–510.

Tang, X., & Wang, X. (2004). Face Sketch Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, *14*(1), 50–57.

Walther, D. B., Chai, B., Caddigan, E., Beck, D. M., & Fei-Fei, L. (2011). Simple line drawings suffice for functional MRI decoding of natural scene categories. *Proceedings of the National Academy of Sciences of the United States of America*, *108*(23), 9661–6.