

A Machine Learning Approach for the Online Separation of Handwriting from Freehand Drawing

Danilo Avola¹, Marco Bernardi², Luigi Cinque^{2(✉)}, Gian Luca Foresti¹,
Marco Raoul Marini², and Cristiano Massaroni²

¹ Department of Mathematics, Computer Science, and Physics, University of Udine,
Via Delle Scienze 206, 33100 Udine, Italy

{danilo.avola,gianluca.foresti}@uniud.it

² Department of Computer Science, Sapienza University,
Via Salaria 113, 00198 Rome, Italy

{bernardi,cinque,marini,massaroni}@di.uniroma1.it

Abstract. The automatic distinction (domain separation) between handwriting (textual domain) and freehand drawing (graphical domain) elements into the same layer is a topic of great interest that still requires further investigation. This paper describes a machine learning based approach for the online separation of domain elements. The proposed approach presents two main innovative contributions. First, a new set of discriminative features is presented. Second, the use of a Support Vector Machine (SVM) classifier to properly separate the different elements. Experimental results on a wide range of application domains show the robustness of the proposed method and prove the validity of the proposed approach.

Keywords: Domain separation · Handwriting · Textual domain · Freehand drawing · Graphical domain · SVM classifier

1 Introduction

Handwriting and freehand drawing are two modalities of communication that allow people to express concepts and ideas naturally. Each of them supports an ever-increasing number of popular desktop and mobile applications [8,9]. Actually, different fields of the technical design (e.g., mechanical engineering) together with an increasing number of professional applications (e.g., freehand annotation systems [15]) require that users are enabled to perform both handwriting and freehand drawing elements on the same interface with the aim to make their design experience as effective and efficient as possible. This paper describes an SVM classifier based approach for the online separation of handwriting from freehand drawing elements. In particular, the paper presents two main novelties with respect to the current literature. First, a new set of highly discriminative features is presented. Second, an SVM classifier to address this

matter is adopted. Since all the measurements on the strokes that compose a scenario are computationally inexpensive, the system works in real-time without special hardware configuration. The experimental results were supported by 25 persons, i.e., 10 persons for the training set and 15 persons for the evaluation set. The experiments were performed on 6 scenarios: electronic circuits, mind maps, Venn diagrams, use cases, flowcharts, entity-relationship diagrams. The obtained results on the accuracy metric prove that this work is a concrete contribution to the current literature.

The rest of the paper is structured as follows. Section 2 provides an overview of the current state-of-the-art in domain separation. Section 3 describes the proposed method, including the set of features and the SVM classifier. Section 4 reports the experimental results obtained on the application domains. Finally, Sect. 5 concludes the paper.

2 Related Work

The online separation of domain elements is a topic that needs to be further investigated. The majority of the methods in the literature are focused on recognizing one or more domains with respect to a specific application context [14]. Examples of multi-domain sketch recognition are presented in [1, 11]. In [11], a mixture of geometrical features and an extensible set of heuristics are used to identify a set of shapes by a fuzzy logic approach. The solution proposed in [1] can identify shapes through an innovative Bayesian network supported by structural descriptions. Unlike these works, the focus of the present paper regards the domain separation. In the current literature, few works are reported. A first approach for separating text and drawing patterns is presented in [17], where the textual domain is formed by Japanese characters. According to the nature of this vocabulary, each stroke is considered as a set of segments. Instead, the features are based on the relationships between the segment length, the number of segments, and the bounding-box size (i.e., the small rectangle that contains all segments). Following, the method proposed in [6] is based on the Multi-Layer Perceptron (MLP) and Hidden Markov Model (HMM). The MLP performs a text domain recognition on the feature vectors extracted from the strokes, instead the HMM discriminates each stroke of the digital ink into two classes: text and graphics. Another interesting work is proposed in [17], where the sum of the angles formed by two consecutive segments, the ratio between the stroke length and the bounding-box size, and the stroke direction on the x and y axis are considered as features. Differently, in [5], the authors perform a classification between shapes and text strokes, in the context of digital ink, by an entropy measure. The latter is obtained by the internal angles of the stroke, where a high value of entropy represents a text, while a low value is associated to a shape. The work proposed in [3] describes an online framework able to automatically distinguish freehand drawing from handwriting, where an interesting feature, called band-ratio, was introduced. This feature considers the distribution of the stroke points within three specific areas, i.e., top, middle, and bottom, of the bounding-box. The work

in [7] uses the features proposed in [5] and introduces a new feature related to the acquisition by hardware mechanism, i.e., the pressure exerted by the user on the pen to create a stroke. This new set of features is used to perform the separation between text and freehand drawing. More specifically, the authors analysed a wide set of Machine Learning (ML) algorithms, including Bootstrap Aggregating, LADTree, LMT, LogitBoost, MLP, Random Forest, and Sequential Minimal Optimization (SMO), to check the discriminative power of the selected features. Finally, the framework presented in [2] shows different interesting stages to separate and recognize text and graphical symbols. In particular, the authors describe separation stage that uses two processes to detect how many and which objects are performed by users. Subsequently, the framework computes mathematical and statistical relationships on each candidate object to provide a reliable classification. Inspired by different works reported above [2,3], but unlike them, the presented work proposes the use of an SVM classifier to perform the separation task. SVM technique, respect other well-known techniques [5,13], can be considered an optimal solution for binary classification. In domain separation, the distinguishing between text strokes and graphical strokes can be seen as a binary classification problem where the features are considered as points of a hyperplane.

3 The Proposed Method

The definitions and terminologies used in this section are defined in [3]. As shown in Fig. 1, the proposed method is composed of four main stages: pre-processing, feature extraction, machine learning, and domain separation. The first deals with simplifying and aggregating each stroke. The second extracts the different features from a stroke and combines them into a single feature vector. The third adopts a learned SVM to classify each stroke in one of the two available classes: textual domain or graphical domain. Finally, the last provides a feedback to the user in real-time.

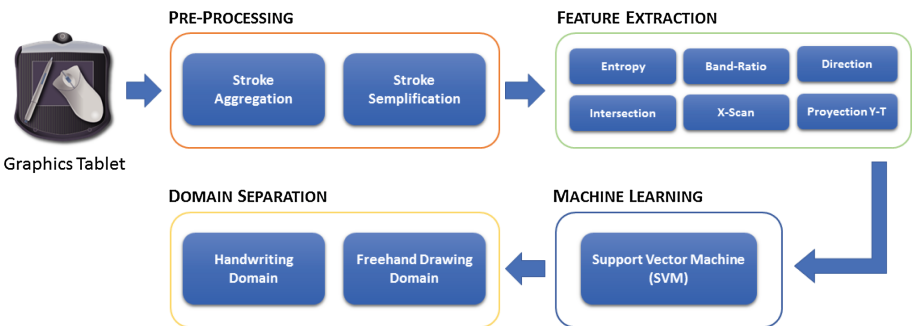


Fig. 1. Logical architecture of the proposed system composed of four main stages: pre-processing, feature-extraction, machine learning, and domain separation.

3.1 Pre-processing Stage

This stage is composed of two processes: *stroke aggregation* and *stroke simplification*. The first detects how many and which strokes must be aggregated. The second simplifies the lines of the stroke by deleting unnecessary points.

Multiple strokes are very frequent in both textual domain and graphical domain. The *stroke aggregation* process checks temporal and spatial relationships among strokes to provide one or more partitioned sets of strokes representing the candidate objects. More specifically, the process examines pairs of consecutive strokes and considers the time interval, linked to each bounding-box, elapsed between the end of the first stroke and the start of the second stroke. To evaluate if two strokes can be aggregated, one of these conditions must be respected:

- if part of a stroke crosses another one, and the areas of the two bounding-boxes have a difference of about 10%;
- if the time interval is less of 500 ms;
- if two strokes are overlapped of about 20%, and at least half of a bounding-box is contained within the other.

At the end of this process, a new set of strokes is created. Then, the latter is sent to the *stroke simplification* process. Often, the strokes are composed of a high number of unnecessary points that may affect the performance and precision of some features. The *stroke simplification* process (or *line fitting*) allows to delete these points thus simplifying curves and lines [12]. In the proposed work, two techniques are implemented [10]: Radial-Distance and Douglas-Peucker. The first (default option) provides an approximation of the elements less accurate but faster ($O(n)$). The second provides a more accurate approximation but with a high computational cost ($O(n^2)$).

3.2 Feature Extraction Stage

Feature extraction is a critical step that can influence the performance of the separation algorithm. In this work, two features, i.e., entropy [5] and band-ratio [2], are inherited by the current state-of-the-art due to their proven usefulness. The other four features have been ad-hoc created to provide a high discriminative feature vector. In this way, a new feature vector composed by six features is implemented (Fig. 2).

Entropy feature is defined in [5] as an accurate criterion to distinguish shapes and text strokes. This feature measures the angles formed by three consecutive points. For each of them, a letter based on its amplitude is assigned. So, each stroke is represented by a string of letters. For each representation of stroke, entropy is calculated as follows:

$$\sum_{x \in X} p_x \log_2 p_x \quad (1)$$

where X is the set of letters, and p_x is the probability that a point is assigned to the letter x .

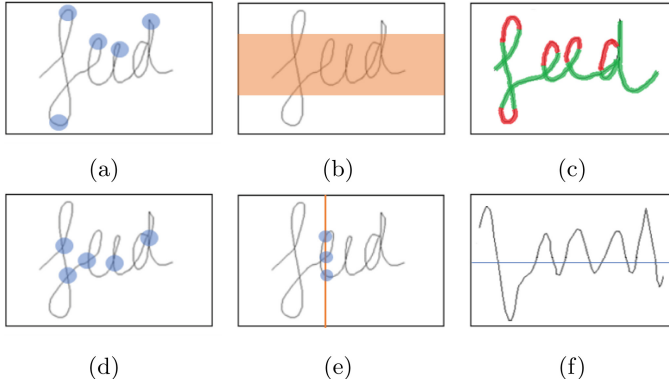


Fig. 2. Set of implemented features: (a) Entropy, (b) Band-ratio, (c) Direction, (d) Intersection, (e) X-Scan, and (f) Projection y-t.

Band-Ratio feature, is defined in [2]. It measures the distribution of the stroke style. This feature is computed from a vertical point, where the band is created. Subsequently, the band is increased until it covers 65% of the points of the whole stroke. The feature is calculated as follows:

$$f_{br} = \frac{h_{band}}{h_{bb}} \quad (2)$$

where h_{band} is the height of the band and h_{bb} is the height of bounding box of the stroke. Its value has a range between 0 and 1.

Direction feature measures represent the number of repeated forwards-backwards movements produced by the stroke. The number of these movements is constant and can be considered a very discriminant measure to distinguish text from drawing. This feature can be calculated as follows:

$$f_d = \frac{\sum_{i=2}^{|N|} |P| S(p_i, p_{i-1})}{l_{bb}} \quad (3)$$

where $|P|$ is the number of the points of the stroke, i is an integer within the interval $2 \leq i \leq |N|$, l_{bb} is the length of the bounding-box, and S is a function defined by the following expression:

$$S(u, v) = \begin{cases} d(u, v) & \text{if } u_x \leq v_x \\ -d(u, v) & \text{otherwise} \end{cases} \quad (4)$$

where u and v are two consecutive points, and $d(u, v)$ is the distance between them. In the case of text, the values are always positive and have a range between 0.1 and 0.5. For the drawing, they can be negative for irregular forms or they can have a range between 0.6 and 1.

Intersection feature measures the number of intersection points of a stroke. The feature calculation process is described as follows:

$$f_I = \sum_{i=2}^{|P|} \sum_{j=i+2}^{|P|} I(p_i, p_{i-1}, p_j, p_{j-1}) \quad (5)$$

where I is a function defined as:

$$I(u, v, w, z) = \begin{cases} 1 & \text{if the segment } uv \text{ intersect the segment } wz \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

X-Scan feature. Given the imaginary vertical segments at regular intervals throughout the length of the bounding-box, the stroke will intersect them many times. The X-Scan feature measures the number of these segments. This features is calculated as follow:

$$f_{xs} = \frac{\sum_{v \in V} I_s(v)}{l_{bb}} \quad (7)$$

where v is the vertical segment considered, l_{bb} is the length of bounding-box, and I_s is a function defined as:

$$I_s(v) = \begin{cases} 1 & \text{if } v \text{ intersects the stroke more than once} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Projection y-t feature measures analyses the horizontal movement of the stroke. To avoid the disturbance of the lateral movement, it operates a data transformation by replacing the x-axis with the acquisition time of the stroke. This transformation produces a sinusoidal-type curve for the text and more irregular patterns for the figures.

3.3 Machine Learning Stage

A good set of features is an optimum starting point, but it is necessary to create or adopt a suitable classifier to reach high level in accuracy and performance. In the proposed context, we have two main factors. The first regarding the natural amount of errors due to the handwriting and freehand drawing activities. The second concerning the binary nature of the matter. These reasons promoted the use of a SVM classifier to estimate, on one side, the values of the different features and their relationships and, on the other hand, to mitigate the propagation of the different errors by a robust hyperplane [4, 16].

3.4 Domain Separation Stage

The domain separation stage manages the Graphics User Interface (GUI) and shows the processing results to the user. In Fig. 3 an example of domain separation is reported.

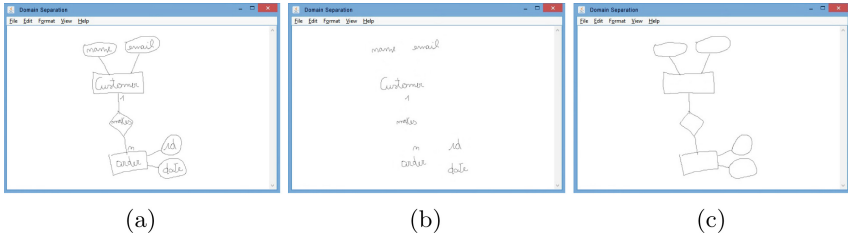


Fig. 3. Online separation GUIs: (a) scenario, (b) handwriting domain, (c) freehand drawing domain.

4 Experimental Results and Discussion

The main purposes of the experiments were the assessment of the set of salient features for the separation between textual and graphical domains, the overall robustness of the proposed approach, and its higher accuracy. The experiments were performed by using a challenging set of scenarios described in Sect. 4.1. A discussion of the results and a comparison of the proposed method with selected key works of the current state-of-the-art are reported in Sect. 4.2.

4.1 Dataset

Nowadays, there is not a common dataset in the field of domain separation. Consequently, in order to show the robustness of proposed approach, a new dataset was built. The dataset is based on the union of the six scenarios used by selected key works of the current literature [2, 5–7, 17]. The scenarios are shown in Fig. 4. From left to right are electronic circuits, mind maps, Venn diagrams, use cases, flowcharts, and entity-relationship diagrams, respectively. These scenarios were chosen for different reasons. First, they allow a comparison with the key works of the current state-of-the-art. Second, they are challenging in domain separation, for example, mind maps is a very difficult scenario because it is not a formalized diagram and each user can have a personal style in drawing the different shapes. In order to train the adopted SVM, a training set was created (in Fig. 5 some instances are shown). In particular, a set of 10 persons aged from 20 up to 30 years, 5 males, 5 females was selected. Each user had to perform, for 8 times, the whole set of graphical symbols represented by the 6 scenarios (an example is provided in Fig. 5a), and for 5 times, a set of summaries of about 1000 words in which the words presented different levels of grouping (an example is provided in Fig. 5b).

4.2 Results

In the evaluation step a set of 15 persons, different from the previous ones (i.e., training step) but with the same characteristics, was selected (9 males, 6 females). To evaluate the experiments, the accuracy metric was adopted [18].

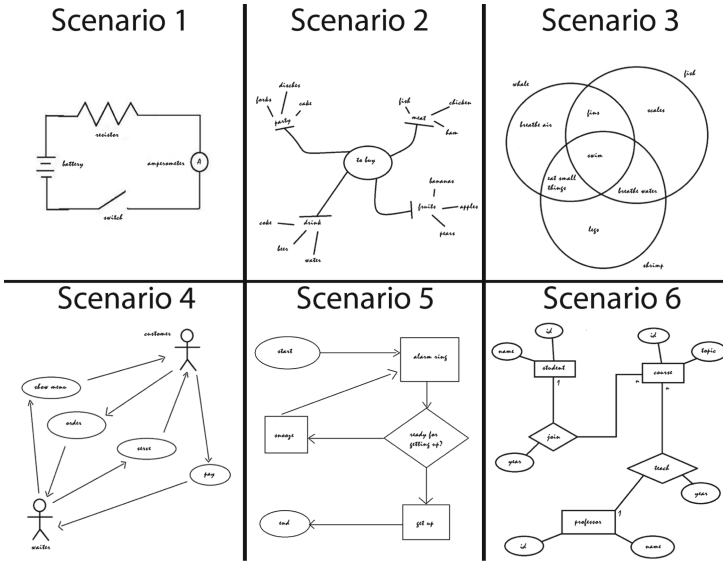


Fig. 4. Scenarios for separation between textual and graphical domains.



Fig. 5. Training set: (a) geometrical shapes, (b) four groups of words with different lengths.

As reported in Table 1, the method achieves an overall accuracy of the 97.3%. In particular, the use cases scenarios has achieved the best accuracy of 98.5%. Instead, electronic circuits and mind maps have obtained an accuracy of 96.5%.

Table 1. Comparison of the accuracy measures among state-of-the-art approaches.

Method	Accuracy
Proposed approach	97.3%
Bishop et al. [6]	97%
Bhat and Hammond [5]	92.1%
Blagojevic et al. [7]	90.5%
Machii et al. [17]	88%
Avola et al. [2]	85%

We have compared the proposed method with five key works of the current literature presented in [2, 5–7, 17]. The comparison is based on the benchmark of these works. In Table 1 the overall results are reported. They show that the proposed method is a concrete contribution to the current literature. As mentioned before, there is not a common dataset to have a direct comparison with these key works. In addition, some of these works, like the proposed one, are based on tests performed by a specific class of users (e.g., young people, computer science students). These factors can influence the experimental phases and often these details about the persons are not present in the other works. Furthermore, different data acquisition methods can be distinguished. Blagojevic et al. [7] use a system similar to that proposed. Instead, Machii et al. [17] and Bishop et al. [6] use an optical system for scanning of strokes. Another consideration regards the handwriting styles. Machii et al. [17] focused on Japanese writing. Instead, Bhat and Hammond [5] and Bishop et al. [6] focused on writing in block letters. All these aspects make the comparison a hard task. To obtain a comparative analysis, we built a dataset containing the contexts in which these works were tested. Blagojevic et al. [7] performed extensive evaluations on diagrams from 6 different domains (4 of these are used into the built dataset). Bishop et al. [6] used data collected among the employees at Microsoft Research in Cambridge, using a purpose-written piece of software and additional tests were also obtained from the Tablet PC Ink Parsing Team at Microsoft in Redmond. Machii et al. [17] used a dataset where they have chosen 18 patterns on which to perform the experiments. Finally, the presented results allow to give two considerations. First, the novel set of features is very discriminating. Second, the SVM is very suitable for this kind of binary separation domain.

5 Conclusions

This paper describes an SVM classifier based approach for the online separation of handwriting (textual domain) from freehand drawing (graphical domain) elements. The paper presents two main novelties with respect to the current literature. First, a new set of highly discriminative features. Second, the use of an SVM classifier. Despite the lack in literature of a dataset and the lack of a standard for the comparison of different approaches in this field, the authors of the present paper have produced wide efforts to provide a reasonable and reliable comparison between them. The experimental tests have provided a high accuracy of 97.3% which shown the concrete contribution to the current state-of-the-art.

References

1. Alvarado, C., Davis, R.: Dynamically constructed bayes nets for multi-domain sketch understanding. In: International Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH), pp. 1–6 (2006)

2. Avola, D., Cinque, L., Placidi, G.: SketchSPORE: a sketch based domain separation and recognition system for interactive interfaces. In: Petrosino, A. (ed.) ICIAP 2013. LNCS, vol. 8157, pp. 181–190. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-41184-7_19](https://doi.org/10.1007/978-3-642-41184-7_19)
3. Avola, D., Del Buono, A., Del Nostro, P., Wang, R.: A novel online textual/graphical domain separation approach for sketch-based interfaces. In: Damiani, E., Jeong, J., Howlett, R.J., Jain, L.C. (eds.) *New Directions in Intelligent Interactive Multimedia Systems and Services - 2*. SCI, vol. 226, pp. 167–176. Springer, Heidelberg (2009). doi:[10.1007/978-3-642-02937-0_15](https://doi.org/10.1007/978-3-642-02937-0_15)
4. Bahlmann, C., Haasdonk, B., Burkhardt, H.: Online handwriting recognition with support vector machines - a kernel approach. In: *Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition*, pp. 49–54 (2002)
5. Bhat, A., Hammond, T.: Using entropy to distinguish shape versus text in hand-drawn diagrams. In: *International Joint Conference on Artificial Intelligence*, pp. 1395–1400 (2009)
6. Bishop, C.M., Svensen, M., Hinton, G.E.: Distinguishing text from graphics in on-line handwritten ink. In: *International Workshop on Frontiers in Handwriting Recognition*, pp. 142–147 (2004)
7. Blagojevic, R., Plimmer, B., Grundy, J., Wang, Y.: Using data mining for digital ink recognition: dividing text and shapes in sketched diagrams. *Comput. Graph.* **35**(5), 976–991 (2011)
8. Chen, Q., Shi, D., Feng, G., Zhao, X., Luo, B.: On-line handwritten flowchart recognition based on logical structure and graph grammar. In: *International Conference on Information Science and Technology (ICIST)*, pp. 424–429 (2015)
9. Degtyarenko, I., Radyvonenko, O., Bokhan, K., Khomenko, V.: Text/shape classifier for mobile applications with handwriting input. *Int. J. Doc. Anal. Recognit.* **19**(4), 369–379 (2016)
10. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. In: *Classics in Cartography*, pp. 15–28 (2011)
11. Fonseca, M.J., Pimentel, C., Jorge, J.A.: Cali: an online scribble recognizer for calligraphic interfaces. In: *Papers from the 2002 AAAI Spring Symposium on Sketch Understanding*, pp. 51–58 (2002)
12. Foresti, G., Murino, V., Regazzoni, C., Vernazza, G.: Grouping of rectilinear segments by the labeled hough transform. *CVGIP: Image Underst.* **59**(1), 22–42 (1994)
13. Foresti, G.L., Micheloni, C.: Generalized neural trees for pattern classification. *IEEE Trans. Neural Netw.* **13**(6), 1540–1547 (2002)
14. Gabe, J., Mark, D.G., Jason, H., Ellen, Y.L.D.: Computational support for sketching in design: A review. In: *Foundations and Trends in Human-Computer Interaction*, pp. 1–93 (2009)
15. Grundel, M., Abulawi, J.: SkiPo - a sketch and flow based model to develop mechanical systems. *INCOSE Int. Symp.* **26**(1), 399–414 (2016)
16. Ma, Y., Guo, G.: *Support Vector Machines Applications*. Springer, Cham (2014)
17. Machii, K., Fukushima, H., Nakagawa, M.: On-line text/drawings segmentation of handwritten patterns. In: *International Conference on Document Analysis and Recognition*, pp. 710–713 (1993)
18. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **45**(4), 427–437 (2009)