Document-Level Positioning of a Pen Tip by Retrieval of Image Fragments

Koichi Kise†, Kazumasa Iwata†, Tomohiro Nakai†,
Masakazu Iwamura†, Seiichi Uchida††, Shinichiro Omachi†††

† Dept. of Computer and Systems Sciences, Graduate School of Engineering,
Osaka Prefecture University, Japan
††Faculty of Information Science and Electrical Engineering, Kyushu University, Japan
†††Graduate School of Engineering, Tohoku University, Japan

Abstract

This paper presents a new method of positioning a pen tip on a document for pen-based computing. Current technologies such as the Anoto pens rely on special mechanisms for positioning, which result in limiting their applicability. Our proposal is to ease this problem — a camera pen that allows us to write on ordinary paper for positioning. A document image retrieval method called LLAH is tuned to position the pen tip efficiently on the coordinates of a document only by capturing its tiny image fragment. In this paper we report results of preliminary experiments about recovering handwriting of simple figures and letters on documents using a prototype camera pen.

1. Introduction

Positioning of a pen tip is a fundamental and important function for pen-based computing. It enables us not only to recover the handwriting (the trajectory of a pen tip), but also to invoke various services such as editing original documents and executing external programs based on the observed position. Services enabled by the positioning depend on the level of positioning. If the user is interested in tracing the pen tip movement, it would be enough to obtain a "local" position, i.e., the displacement relative to the previous position, which is similar to the function of a mouse. On the other hand, if the user would like to invoke the service associated with a document, it is required to obtain a more "global" position, i.e., the position on a page of a document as well as which document the user is working on. We are concerned here with the detection of the global position, which we call the *document-level positioning*.

Several technologies have been developed to obtain the document-level position of a pen tip. There is no doubt that the most famous one is the Anoto technology which employs special paper and a camera pen to obtain the document-level position. On the special paper the global position is encoded by a grid of fine dots. The camera pen can be used as a normal pen and simultaneously decodes the position from a captured configuration of fine dots.

Although the Anoto pen and its equivalents have the above advantage on positioning the pen tip, they also have a limitation that they require special paper to function. They detect not the printed foreground (textlines, etc.) on a sheet of paper but fine dots with which services are associated. This discrepancy may cause a problem that available services may be different even with the same printed foreground. In other words, the user needs to have the same sheet of paper with the identical fine dots to provide the same services.

In order to make the relation between documents and services more simple and intuitive, the pen needs to identify not the fine dots but the printed foreground itself like human beings do. If such a pen is available, we can print documents on any paper to supply services. All we need is to have an intelligent pen that is capable of identifying the printed foreground.

In this paper we propose a camera pen that enables us the above mentioned function of identifying the printed foreground. The position of the pen tip is detected by matching the camera-captured image and the foreground pattern of documents. Since the camera is mounted on the pen, the captured area is limited and suffers from the geometric distortion caused by a skewed camera position to a sheet of paper, both of which make the matching nontrivial.

The proposed method solves the above problem by applying the document image retrieval called LLAH (locally likely arrangement hashing) [4]. The LLAH allows us to find which document and where in the document the pen tip is located. We also report results of preliminary experi-

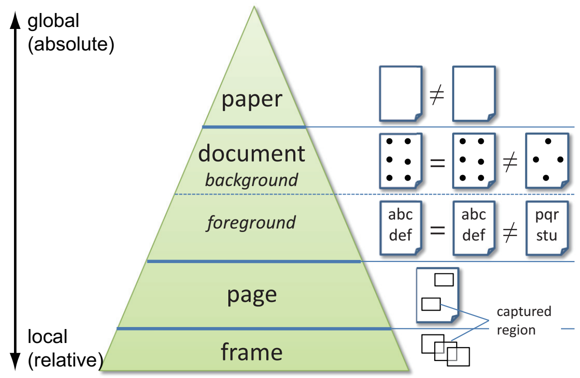


Figure 1. Levels of positioning.

ments on recovering handwriting of simple figures and letters. Although we have found several problems to be solved for its actual use, the potential ability of LLAH for detecting pen tip positions has also been uncovered.

2. Pen Tip Positioning

First of all, let us consider how to detect the position of a pen tip with the help of imaging devices. Figure 1 shows various levels of detection, where lower (upper) parts represent more local (global) positioning.

At the lowest level, we indicate local positioning by matching images of subsequent frames. The matching allows us to detect the displacement, i.e., a relative position of the pen tip from the previous frame. Optical mice function based on this principle.

A drawback of this "optical mouse" approach is that it cannot identify reappearance of previously seen regions. For example, once the starting point of drawing a circle disappears from image frames, it is not easy to identify it when it reappears in image frames. Ultrasonic pens and tablets can solve this problem by directly measuring the position of the pen tip, though they are not imaging devices. A problem of such devices is that they do not detect the pen tip position on a page but measure the position relative to a special device such as an ultrasonic receiver. Thus if the special device shifts on a document by accident, correct position cannot be detected anymore. Methods using a video camera [3, 6] attempts to remove such a special device by using visual input. However, no position on a page can similarly be detected; what is detected is still the position relative to the video device.

This problem may be solved by extracting characteristic features from the surface of a sheet of paper. These features are stored and matched to the current one to know whether the current image fragment has already been seen. Uchida et al. are working on this approach [7].

If the user is interested in detecting not the position on

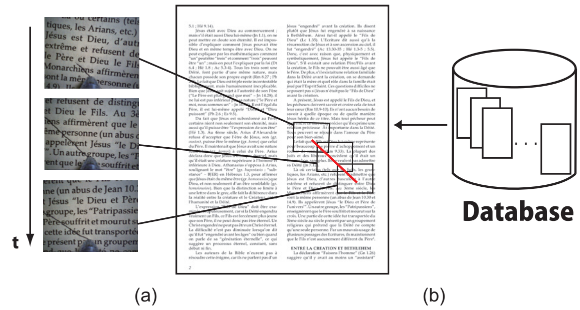


Figure 2. Task of positioning the pen tip.

a page, but the position on a document, i.e., on which document and on which region of a document the pen tip is located, a different approach is required. The most famous one is the Anoto pen which utilizes special paper with dots on its background for locating the absolute position of the pen tip. However, the requirement of special paper limits its applicability. It is also worth noting that the pen detects not a region in a document but dots on a sheet of paper. Thus, if different documents are printed on the sheets of paper with the same dots, they cannot be distinguished.

Such a problem would likewise be solved by measuring foreground (printed pattern) of documents. This approach takes printed patterns of the document foreground as landmarks to detect the pen tip position. Since the landmarks are obtained from documents themselves, the same position can always be detected on the same document. In this paper we pursue this approach.

The above two approaches achieves positioning at the document level by observing the foreground or background of documents. Sheets of paper with the same foreground or background are identified as the same document. A different way of pen tip positioning is by distinguishing sheets of paper, which we call the paper-level positioning. If characteristic features called paper fingerprints can be extracted and stored from every region of all sheets of paper, they can be used to know on which sheet the pen tip is. However, the current technology requires a high resolution scanner to extract the paper fingerprints [1]. It is not directly applicable to pen-based computing.

3. Task and Approach

3.1. Task definition

The task of positioning the pen tip in this paper is by using a document foreground. The basic idea is quite simple. We assume that a camera is fixed on the pen and thus the pen tip is always at the fixed position of captured images. This means that, as shown in Fig. 2 (a), the task of positioning the pen tip is equivalent to locating the image captured by the camera on the document coordinates.

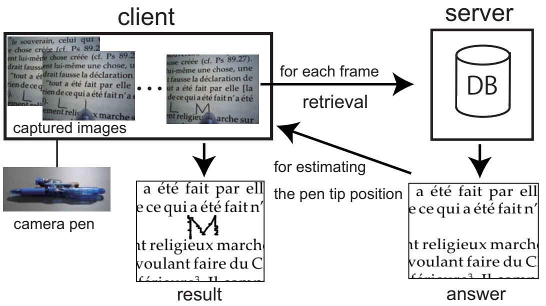


Figure 3. System architecture.

We assume that documents to be annotated are machine-printed and have already been stored as images in the database. Since most documents are printed from their electronic version in recent years, we think this assumption is not unrealistic. Under this assumption, the above mentioned task of locating captured images is equivalent to retrieving their corresponding documents and their parts from the database of document images as shown in Fig. 2(b).

In this paper we exclude from our task the case that the user writes on blank paper. Although this case needs further development, we think that we can apply the approach by Uchida et al. [7].

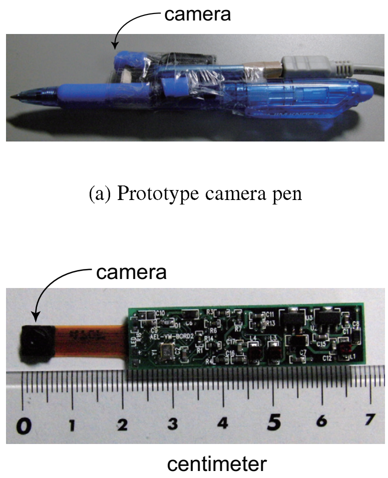
3.2. System configuration

As shown in Fig. 3, the proposed system is based on the client-server model: The camera pen is connected to the client for processing of captured images. The result of processing is sent to the server for finding the corresponding part from the stored document images. The result of retrieval at the server side is returned to the client for keeping track of the pen tip movement on the coordinates of the retrieved document.

Figure 4(a) shows the prototype of our camera pen. The camera module mounted on the pen (called "pen camera" in the following) is capable of capturing video data with the VGA resolution, 30 frames/s., and its viewing angle is 52°. The pen camera and the client computer are currently connected through USB but it can be replaced by wireless connection. The pen camera is shown in Fig. 4(b) in which the size is shown in [cm].

3.3. Overview of the processing

The key processing of our method is to locate the captured image fragment on the coordinates of a document. A simple solution would be the template matching of image fragments. However it is strictly prohibitive due to its computational cost. Therefore, a fast and accurate method of image matching is mandatory for our approach. We have



(a) Prototype camera pen

(b) Camera module

Figure 4. Prototype camera pen and its camera module.

solved this problem by adopting a method of document image retrieval called LLAH (Locally Likely Arrangement Hashing) with some improvements, which are described in the next section.

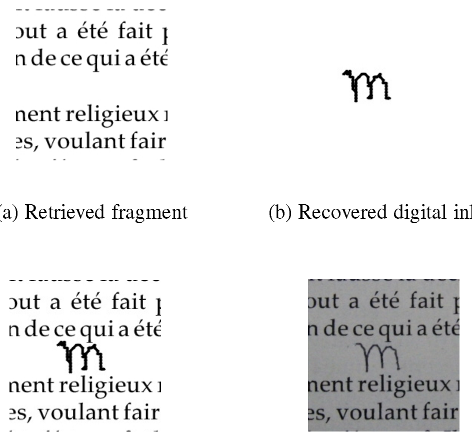
Since the pen tip is always fixed in captured image fragments, it is straightforward to find its position. LLAH tells us not only the matched region, but also values of parameters of projective transformation. Using these values, we can locate the pen tip on the document coordinates. Figure 5 illustrates an example of recovered handwriting. By connecting a sequence of the coordinates of the pen tip for retrieved fragments (Fig. 5(a)), we can recover the handwriting (Fig. 5(b)). The recovered handwriting can be recorded on the original document (such as PDF) as shown in Fig. 5(c). Although there is still some jagged noise on the recovered handwriting, it looks similar to the annotation written on a paper document shown in Fig. 5(d).

4. Improvement of LLAH

This section describes the improvement of LLAH that enables us to track the pen tip from a sequence of camera-captured images.

4.1. Problems of the use of LLAH

LLAH is a method of fast and accurate retrieval of document images which takes as queries camera captured fragments of documents. It was first proposed for documents written in Latin scripts [4] and recently extended to be applicable to many other scripts [5]. Since it is intended to re-



(a) Retrieved fragment (b) Recovered digital ink

(c) Superimposed ink on the electronic document (d) Handwriting on the paper document

Figure 5. Example of recovered handwriting.

trieve documents, the sizes of regions acceptable as queries are relatively large: from whole pages down to 1/8 pages.

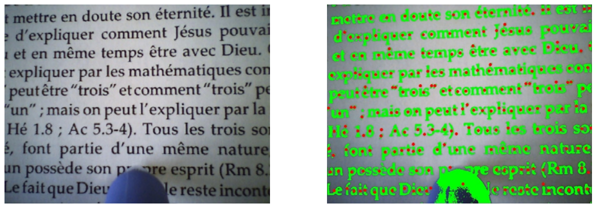
However, the size of fragments captured by the pen camera is limited due to the restriction of its position on the pen. Currently about 10 textlines can be captured. This is clearly much smaller than the size required by LLAH. The application of LLAH to the camera pen, therefore, needs its improvement.

4.2. Feature points

LLAH indexes documents using feature points extracted from their images. For documents written in Latin scripts, centroids of word regions are originally employed as feature points for the purpose of document image retrieval. Their local arrangements defined with nearby 6–10 points are unique enough to distinguish a feature point among more than 10 million. The indexing and matching of feature points are not perfect and thus not all feature points are correctly matched. Nevertheless documents can be retrieved with high accuracy because of the effect of voting with many feature points.

Images captured by the pen camera include feature points whose number is much smaller than that for document image retrieval. Thus direct application of LLAH to the camera pen fails to locate image fragments.

In order to solve this problem, we employ not word regions but connected components from which feature points are extracted as their centroids. An example is shown in Fig. 6. As shown in this figure, connected components allow us to obtain many feature points required for the retrieval.



(a) Captured image (b) Connected components and their centroids

Figure 6. Feature points extracted from a captured image.

4.3. Indexing

In LLAH, each feature point is indexed using multiple feature vectors calculated from surrounding feature points. These feature vectors represent local arrangement of points. Figure 7 shows the overview of the calculation.

The central point surrounded by the points 1–6 is a point of interest to which a feature vector is now calculated. First, note that the point between the numbers 4 and 5 is not included in the feature calculation. This is to cope with disappearance of points in a query image. In the case of Fig. 7, we take 6 points out of 7 points nearest to the point of interest. This enables us to have a feature vector robust to disappearance of one point. Since we have 7 ways of selecting 6 points from 7 points, and each combination corresponds to a feature vector for indexing, the point of interest is indexed in 7 different ways.

For each combination of 6 points, we take all combinations of 4 points out of 6 points. From each of 4 point combinations, we calculate an affine invariant defined as the area ratio of two triangles as shown at the lower left corner in Fig. 7. The affine invariant is used to form a feature vector whose dimension is 15 (no. of combinations of 4 out of 6). This is the basic indexing by LLAH.

The above features are, however, not sufficient for distinguishing a tiny image fragment. In order to improve the discrimination power of the feature vector, we add some more dimensions as shown at the lower right corner of Fig. 7. Additional features are obtained by taking into account the area of connected components. To be precise, ranks of the area of connected components are newly added as features, since the ranks are fairly stable with the variation of images.

The total number of dimensions of a feature vector is 21 ($= 15 + 6$). The number of feature points is about 3K–4K / page, and thus the number of feature vectors for indexing one page is about 20K – 30K.

In the database, document images are indexed by using a hash table that stores quantized feature vectors with a label of document ID.

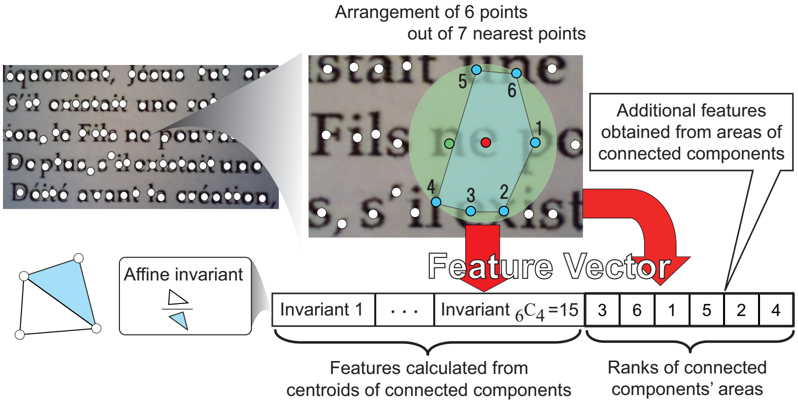


Figure 7. Calculation of feature vectors.

4.4. Retrieval

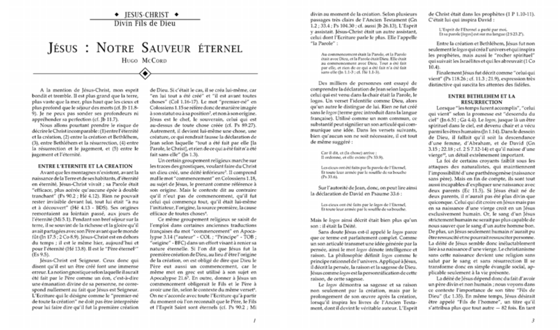
The processing of retrieval is similar to the indexing. Each captured image is employed as a query image to the database. First, feature vectors are calculated for all feature points extracted from the captured image. Each feature vector is utilized to find its nearest neighbor (NN) from the database. Because the number of feature vectors stored in the database is so large that the sequential matching is prohibitive. Hashing helps us to drastically cut the time for retrieval. As a result of looking up the hash table, each feature vector from the query corresponds to a feature vector in the database. This point-wise correspondence is considered as a vote for a document. The document which has the maximum votes is regarded as the result of retrieval. Although the retrieved feature vector may not be the "real" nearest neighbor, it causes few problems due to the effect of voting.

Once the retrieved document is determined, the next step is to find in the document the region to which the query image corresponds. This is achieved by using again the correspondence of feature points. We apply RANSAC [2] to make the estimation of homography reliable.

As the result of estimation, the location of the pen tip is estimated as a fixed position in the region projected onto the document coordinates. We can simply recover the handwriting by connecting the estimated position at each frame of video data.

5. Preliminary Experiments

The basic functions of the proposed pen are (1) to retrieve on which document the pen tip is, and (2) to detect where in the page the pen tip points. Since the pen captures video data, results should be evaluated for each frame image of the video as well as for the whole sequence of the writing. We evaluated the proposed pen from the above point of view using the following data.



(a) (b)

Figure 8. Examples of original documents for annotation.

5.1. Data

The database of documents for matching consisted of 1,200 pages written in 12 languages including Spanish, French, English, Chinese and Japanese. The size of images varies between 3K $\times$ 5K and 5K $\times$ 7K. As the documents for annotation, we selected 10 samples from Spanish documents and also 10 from French, whose examples are shown in Fig. 8.

As annotations, simple figures such as lines, circles, triangles, rectangles, and pentagrams, as well as a few letters such as A, ab, S, L, N, etc. were written on printed foreground and margin between textlines and columns. The camera mounted on the pen can catch about 10–12 textlines.

5.2. Results and Discussions

The results of retrieval and the quality of recovered handwriting are summarized in Table 1. In this table, the number

Table 1. Experimental results.

id	# frames	retrieval rate	quality
French 1	306	83.0%	G
2	290	97.2%	G
3	231	91.8%	G
4	281	69.4%	F
5	230	87.4%	F
6	253	77.1%	G
7	212	18.4%	NG
8	195	92.3%	F
9	201	88.6%	F
10	191	70.2%	F
Spanish 1	267	98.5%	G
2	319	100%	G
3	397	84.1%	G
4	412	42.7%	F
5	286	97.6%	G
6	216	63.0%	G
7	261	85.1%	G
8	477	77.8%	G
9	362	89.8%	G
10	241	0.4%	NG

of frames indicates the number of query images for the retrieval and the retrieval rate is the rate of correct retrieval. Since we did not employ rejection in this experiment, retrieval errors were almost with erroneously retrieved images [1]. As the quality, we subjectively evaluated the quality of recovered handwriting at three levels: good (G), fair (F) and no good (NG). The recovered and the scanned handwriting are shown in Figs. 9 and 10 for French and Spanish, respectively.

First, there is a clear relationship between the retrieval rate and the quality of recovered handwriting.

For handwriting with the quality G, the shape is almost recovered correctly though it is often with some jagged parts. It is simply because we did not apply smoothing but simply connected positions of the pen tip. For those with the quality F, some parts of handwriting are missing due to the failure of retrieval. This problem is caused by our simple strategy of retrieving each frame independently. Thus by taking into account that a sequence of frames are for the same document, the quality could be improved. To be precise, the pen tip positioned on a wrong document can be corrected by relocating it on the right document, which can be estimated by the voting. For the case with NG, retrieval was failed for most frames. This type of failure was observed if the pen was on a wider white area or on sparse textlines. This indicates the limitation of the current LLAH: LLAH needs connected components as landmarks for positioning the pen tip. We need to compensate this limitation by taking into account other source of information for tracking the pen tip such as proposed in [7].

[1] In the case of no vote for all images, no result is returned. We also had such cases.

6. Conclusion

We have presented a prototype camera pen which works on ordinary paper to recover handwriting. The characteristic point of our method is the use of document foreground patterns as landmarks for positioning the pen tip. The novel algorithm called LLAH is effective to find the position by retrieving the captured image fragments. From the preliminary experiments, we have confirmed the effectiveness of our prototype, though we still have room for further improvement. One is to make the recovered handwriting smoother. Another open problem is to extend the method to work on blank paper. They will be solved in the future work.

Acknowledgment

This work was supported in part by the Grant-in-Aid for Scientific Research (B)(19300062, 20300049) and the Grant-in Aid for Exploratory Research (21650026) from Japan Society for the Promotion of Science(JSPS).

References

[1] W. Clarkson, T. Weyrich, A. Finkelstein, and N. Heninger. Fingerprinting blank paper using commodity scanners. In *Proceedings of IEEE Symposium on Security and Privacy*, 2009. (to appear).

[2] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24:381–395, 1981.

[3] M. E. Munich and P. Perona. Visual input for pen-based computers. *IEEE Trans. PAMI*, 24:313–238, 2002.

[4] T. Nakai, K. Kise, and M. Iwamura. Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. In *Lecture Notes in Computer Science (7th International Workshop DAS2006)*, volume 3872, pages 541–552, 2006.

[5] T. Nakai, K. Kise, and M. Iwamura. Real-time retrieval for images of documents in various languages. In *Handout of 8th IAPR International Workshop on Document Analysis Systems (DAS2008)*, page 37, 2008.

[6] J.-H. Seok, S. Levasseur, K.-E. Kim, and J.-H. Kim. Tracing handwriting on paper document under video camera. In *Proc. of Int'l Conf. on Frontiers in Handwriting Recognition*, 2008.

[7] S. Uchida, K. Itou, M. Iwamura, S. Omachi, and K. Kise. On a possiblity of pen-tip camera for the reconstruction of handwritings. In *Proceedings of CBDAR2009*, 2009.

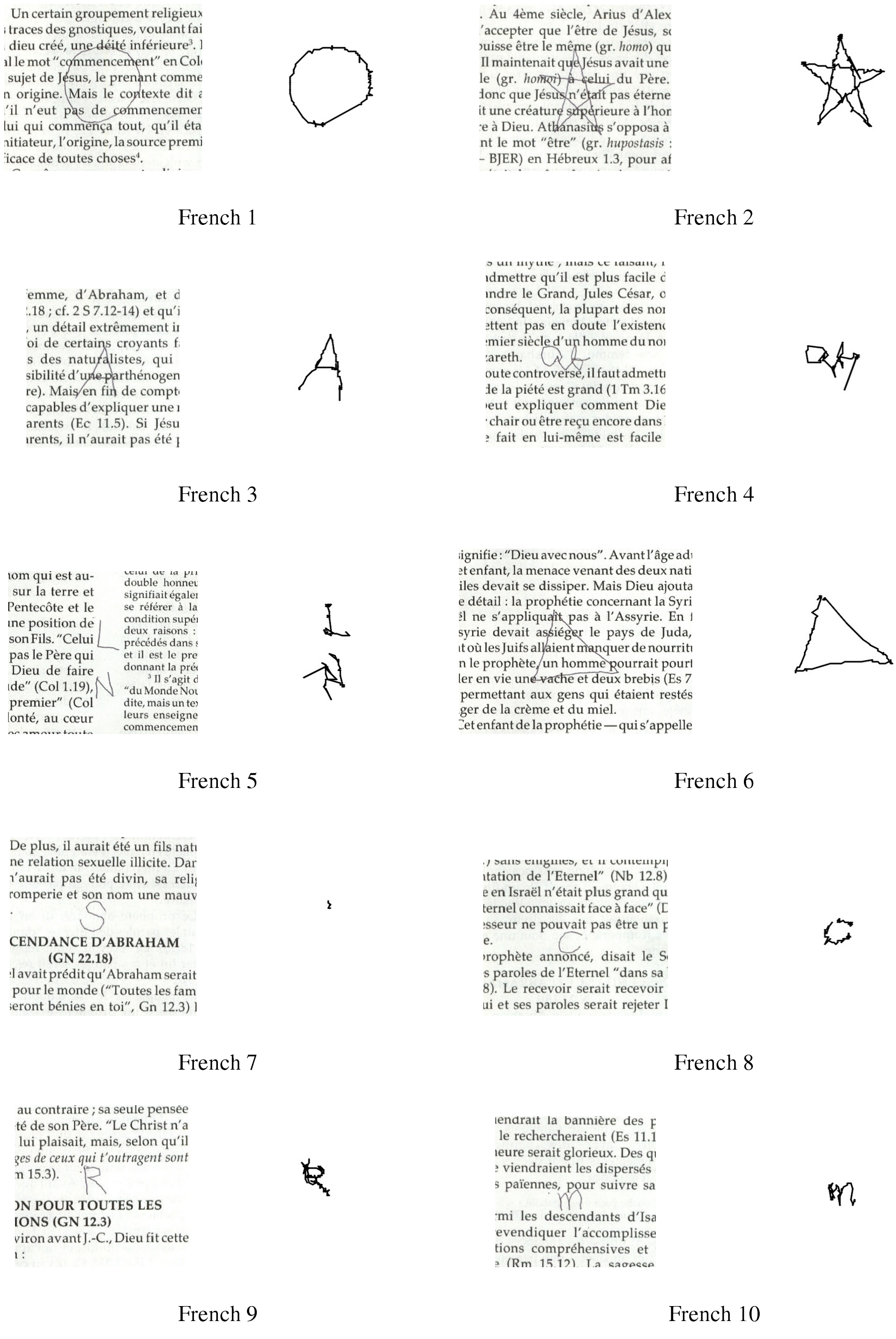


Figure 9. Recovered handwriting for French documents.

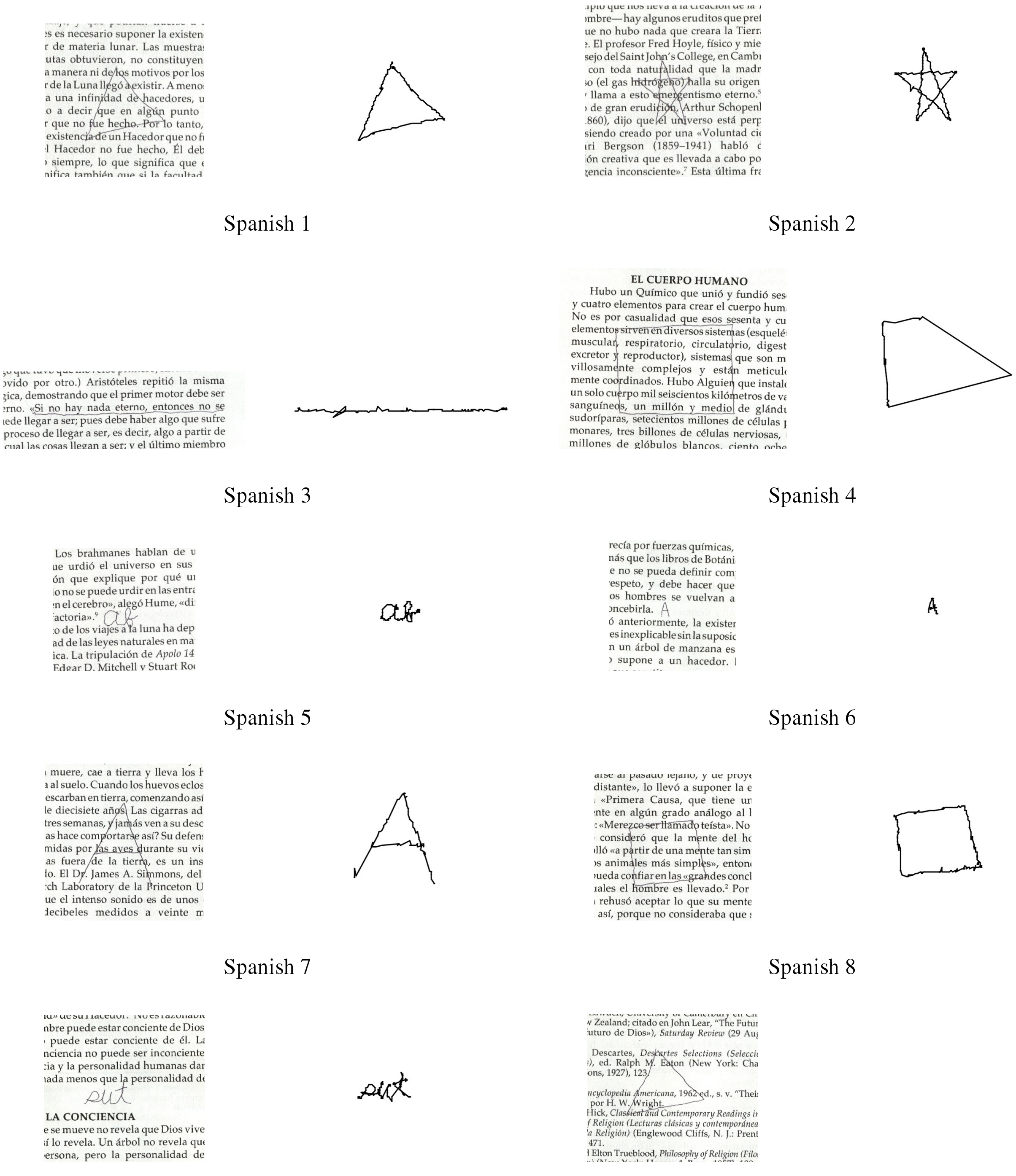


Figure 10. Recovered handwriting for Spanish documents.