ANCHOR POINT SELECTION BY KL-DIVERGENCE

Author(s) Name(s)

Author Affiliation(s)

ABSTRACT

Selecting anchor points for the identification of scanned documents can be an effective and quick means of identifying unknown documents. Here, we discuss and compare some strategies for classification of scanned forms using anchor points and show experiments indicating that a robust system can be built with only a few training examples.

***Index Terms*—** Document Fingerprinting, Anchor Point Selection

1. INTRODUCTION

Consider the following problem: A small business receives or generates several hundred documents per day, all of which must be organized and stored. With a high-speed scanner, this business can, with only a small amount of time and inconvenience, convert all of these documents to electronic form for archiving and ease of access. Unfortunately, this solves only a small part of the problem; The documents have been moved from the analog to the digital domain, but each of them must still be looked at, categorized, and moved to the proper location.

A helpful fact is that, for a given business, many of the documents they scan may look largely the same. A given business will likely receive bills from the same suppliers each month, and these will be the same other than some of the writing on the bill. Collections of legal documents, government forms, and medical forms will often follow the same pattern. Is it possible, given a scanned document, to identify it as a given type, having seen some small number (say, less than ten) of the same type of document?

This problem is not new in the computer vision literature. Near-duplicate detection [1] has seen extensive and is a solved problem in many important cases. This problem is somewhat different as filled in forms may not be near-duplicates of one another, depending on the amount of information in the form. OCR-based approaches [2] are common, but are generally expensive to license and are computationally complex. In addition, the approach we discuss below is far more efficient and can be used in a "cascade" with an OCR-based approach to provide a faster, more robust system than either one alone.

By far the most relevant work to the work presented here, however, is the work by Sarkar [3] in automatic anchor point selection. In this work, anchor points (described in the following section) are chosen by examining matching and non-matching documents, and choosing the best points based on the ROC curve implied by each point. The use of the area under the ROC curve, however, has largely been discredited as a measure of accuracy [4]. Furthermore, Sarkar's work shows no comparisons to other methods, as we do below.

2. MATCHING VIA ANCHOR POINT COMPARISON

A common technology for matching documents is by using *anchor points* [5]. Typically, a human operator will use software to designate several small parts of a form as anchor points. When a new document is scanned, the corresponding parts on the scan will be compared with designated points. If the points on the scanned document are sufficiently similar to the points on the form, the document is classified as being a form of this type, and filed appropriately.

This approach has several benefits: Because only a small number of points are chosen, a document can be classified very quickly. In addition, the small number of points makes the approach robust to defects that obscure much or even most of the scanned form. Third, if the points are chosen correctly, the system can classify with fairly high level of accuracy (as we will see later on).

The last benefit calls attention to the major problem with the approach, namely, the presence of the human operator. It is unlikely that the end user of such a system will be able to choose suitable anchor points, and hiring a technician to do this for each document in the database is economically undesirable because of the level of expertise involved. Specifically, the operator of the system needs to choose anchor points that meet two criteria:

1. The points must have fairly low variability in all instances of the scanned document. That is, the points cannot contain any part of any of the form input fields, as they will be different on every copy of the completed form.

2. The points must have as little in common as possible with the corresponding points on all documents in the database that are *not* of the given type. The more *dissimilar* the anchor points from the corresponding points on non-matching documents, the easier it will be to

tell the difference between matching and non-matching documents.

In the next section, we hope to automate this process of choosing anchor points by examining both the matching and non-matching sets of documents to determine those which discriminate best between the two sets.

3. AUTOMATICALLY CHOOSING ANCHOR POINTS USING THE *KL-DIVERGENCE*

Our approach will proceed, briefly, as follows: We assume that we have a small set of matching documents (that is, documents that are all of the same type), and also a much larger set of documents that are at least mostly composed of documents not of the type in the small set. For each possible anchor point, we will represent that point in the document as a vector. The distance between each document and the reference document with respect to an anchor point can be represented as the Euclidean distance between the corresponding vectors. Thus, we can compute the mean and variance of the distances between corresponding anchor points within the small set. We can do the same for the larger set by comparing each document in the larger set to a reference image of the document from the small set[1]. This will give us two Gaussian distributions: The higher-mean Gaussian corresponding to the collection of non-matching documents, and the lower one corresponding to the matching documents.

We then compute the *KL-Divergence* between the two Gaussians. The KL-divergence D_{KL} between two distributions $p(x)$ and $q(x)$ is defined as:

$$D_{\text{KL}}(P||Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx \qquad (1)$$

If we assume that $p(x)$ and $q(x)$ are normal distributions, this quantity can be easily computed [6]. In general, the KL-divergence is non-symetric, and so we define KL_{max} as:

$$KL_{\text{max}} = \max(D_{\text{KL}}(P||Q), D_{\text{KL}}(Q||P)) \qquad (2)$$

This will help us lower bound the performance of the system below.

We can now compute the KL_{max} for each anchor points, thus getting a measure of how *discriminative* each anchor point is with respect to the collection of non-matching documents. We use this ranking to choose the anchor points in our experiments below.

4. EXPERIMENTS

For these experiments, we use a database of 1007 tax forms downloaded from the IRS website, as well as a database of 1588 "junk mail" documents such as catalogs, magazines, and direct mailings. To simulate form data, we chose one form at random from the tax form database and had several volunteers fill it out by hand. These documents were then scanned on a Kodak 1220 high-speed scanner. All documents were either downsampled or upsampled to 1224×1583 pixels before matching. The documents were then horizontally and vertically registered using the techniques in [7].

As possible anchor points we consider all 30×30 windows in the document, excluding those less than one inch from any edge. We represent each window as a 25 element vector by breaking it down into 25 6×6 sub-windows. Each element of the vector is calculated by multiplying the pixels in each sub-window by a 2-d hamming function, significantly lowering the strength of the pixels near the border of each window. We do this to lessen the effect of any error in registration the was not corrected in the registration phase of the system. Following this multiplication, all of the points in the sub-window are summed. Doing this for every sub-window gives the desired 25-element representation.

We then select 20 anchor points in three ways: The first is to select points that have the best compromise between maximizing the intra-vector variance while minimizing the distance between these vectors in the matching set of documents. That is, points are selected based on the amount of graphical "action" in the sub-windows and whether or not they differ in the matching documents. This represents a simple approach based on intuition. Our other two methods use the technique outlined in Section 3 to select the points. The first of these simply chooses the 20 best points according to KL-divergence. The second will choose a single point, then recompute the distance between documents before selecting the next point, conducting a greedy optimization of the KL_{max}

4.1. Results

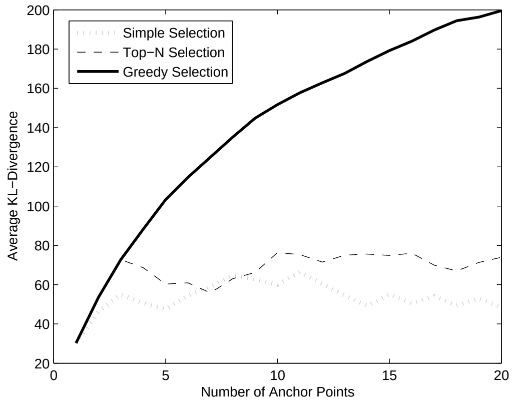

Fig. 1. Curves showing the performance of the three methods with increasing numbers of anchor points.

[1] Ideally, this reference image should be an average of the images in the small set, but a random example from the small set could serve as well.

Our primary result is the graph in Figure 1. The curves plot the average $KL_{\max}$ over the two collections, using the filled forms as the collection of "matching" documents. In general, increasing the number of anchor points should also increase the $KL_{\max}$ between the positive and negative sets of documents. Higher is better.

As we can see, the N-best approach outperforms the intuitive approach as the number of anchor points increases, but the greedy optimization far outperforms both of the other approaches. We thus see that selecting anchor points greedily to maximize our performance measure has a dramatic effect on our level of success. We have also demonstrated the utility of taking into account the non-matching documents in the collection when choosing anchor points.

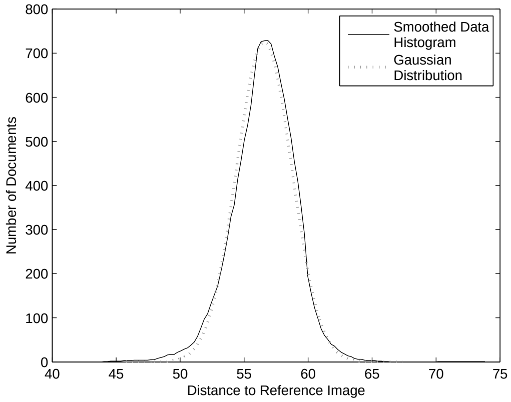

Fig. 2. Kernel smoothed histogram of document distances with a Gaussian distribution fit to these distances.

One concern may be that the assumption of the normality of the distribution of distances between documents is significantly violated, so that our computation of the KL-divergence is invalid. In Figure 2 we plot the Gaussian distribution of the distances between the reference image and all non-matching images. On the same plot is a kernel-smoothed histogram of those distances. As we can see, the correlation between the two curves is very high.

Finally, in Figure 3 we show the anchor points chosen on the reference form in three cases: From left to right, the first is the points chosen by the simple method. The second is the points chosen by the greedy method when the non-matching documents are the set of tax forms. The third is the points chosen by the greedy method when the non-matching documents are the set of junk mail documents. We see that the simple method chooses several points that are very close together and look very similar. While choosing one of these points may be useful, choosing many is unlikely to have additional discriminative effect.

When choosing points against the tax form data, the greedy algorithm selects points in the right-center text of the document, a location where there is little text in the rest of the data. On the junk mail data, the algorithm shows a higher preference for axis-parallel edges, something that is less common in the junk mail data than it is on the tax form data.

4.2. A Comment on Performance

We have arrived at a superior method of choosing anchor points by essentially conducting a greedy optimization of the KL-divergence. But how relevant is the KL-divergence to the performance of the system? To understand how the $KL_{\max}$ metric relates to retrieval performance, consider two gaussians, with μ_p and μ_n representing the mean distances for the positive (matching) and negative (non-matching) classes, respectively, and σ_p^2 and σ_n^2 representing the variance of those distances. For some value x we can represent the ratio of the distances of positive to negative documents as $p(x)/n(x)$ where $p(x) = \mathcal{N}(x; \mu_p, \sigma_p)$ and $n(x) = \mathcal{N}(x; \mu_n, \sigma_n)$.

This ratio represents an important performance parameter, the *true positive rate*, which is the probability that the document matches given that the system predicted it does match. If the documents were uniformly distributed, we could get a reasonable estimate of the log of this ratio by averaging it at several points below the decision threshold:

$$\frac{1}{n}\sum_{b}^{t} \log \frac{p(x)}{n(x)}$$

where n is the number of samples and b is some value much less than the decision threshold t.

The points are not uniformly distributed, however; Their distribution is controlled by the sum of $p(x)$ and $n(x)$ at any given x, normalized by some constant, C. With this, we can use an integral to compute the average:

$$\int_{-\infty}^{t} \frac{p(x) + n(x)}{C} \log \frac{p(x)}{n(x)} dx$$

Let us assume that we choose a prediction threshold t, and for values $x < t$, $n(x) + p(x) \approx p(x)$. That is, assume that the Gaussians are "well-separated". In this case, we can approximate the integral as follows:

$$\int_{-\infty}^{t} p(x) \log \frac{p(x)}{n(x)} dx$$

We are now actually quite close to the definition of the KL-divergence, only differing in the limits of the integral. However, in many cases, the KL-divergence turns out to be a lower bound for this integral. Specifically, consider that the KL-divergence can be written as follows:

$$\int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{n(x)} dx =$$
$$\int_{-\infty}^{t} p(x) \log \frac{p(x)}{n(x)} dx + \int_{t}^{\infty} p(x) \log \frac{p(x)}{n(x)} dx$$

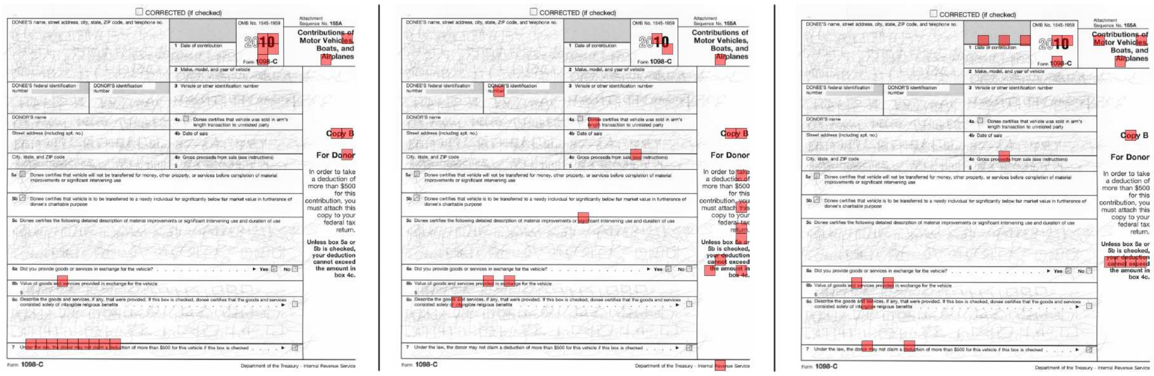
CORRECTED (if checked)
Contributions of Motor Vehicles, Boats, and Airplanes
2010
Form 1098-C
Copy B
For Donor
Form 1098-C
Department of the Treasury - Internal Revenue Service

Fig. 3. Examples of anchor points chosen by, from left to write, the simple method, the greedy method on the tax form data, and the greedy method on the junk mail data.

Let us choose t to be anywhere in between the means of the two Gaussians, so that $t > \mu_p$, $t < \mu_n$. Given that Gaussians are symmetric:

$$\int_t^\infty p(x)dx < \int_t^\infty n(x)dx$$

therefore,

$$\int_t^\infty p(x)\log\frac{p(x)}{n(x)}dx \le 0$$

and thus:

$$\int_{-\infty}^\infty p(x)\log\frac{p(x)}{n(x)} \le \int_{-\infty}^t p(x)\log\frac{p(x)}{n(x)}$$

It then follows that the KL-divergences can be seen as an lower bound on the true positive rate for the system. A mirror-image proof can be constructed showing the same case for the true negative rate. Using KL_{max} as our performance measure essentially means we are reporting the worse of the two numbers, and thus placing a lower bound on the overall performance. In cases where the Gaussians overlap significantly, however, this bound will not be very tight. The bound becomes more and more tight as the Gaussians separate further.

Thus, the KL-divergence is an appropriate criterion for optimization in such as system, as it relates directly to the retrieval performance.

5. CONCLUSIONS

We have here outlined a system that chooses anchor points for document retrieval. Through experiment, we showed that this system is able to match scanned forms with high accuracy, given a training set of only ten completed forms. In addition, we showed that the system's accuracy is far better when the user's non-matching data is used to help train the system. Finally, we showed that the KL-divergence, our optimization criteria for the system, is both easily computed and highly relevant to document retrieval. Future work obviously includes testing on larger document collection and testing the system's response to various quality issues in the scanning process.

6. REFERENCES

[1] Yang Hu, Mingjing Li, and Nenghai Yu, "Efficient near-duplicate image detection by learning from examples," in *ICME*, 2008, pp. 657–660.

[2] Xiaofan Lin, "Ddr research beyond cots ocr software: a survey," in *IS&T/SPIE Conf. on Document Recognition and Retrieval XII*, 2005, pp. 16–20.

[3] Prateek Sarkar, "Learning image anchor templates for document classification and data extraction," in *ICPR*, 2010, pp. 3428–3431.

[4] David J. Hand, "Measuring classifier performance: a coherent alternative to the area under the roc curve," *Machine Learning*, vol. 77, no. 1, pp. 103–123, 2009.

[5] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*, John Wiley & Sons, 2009.

[6] Jacob Goldberger, Shiri Gordon, and Hayit Greenspan, "An efficient image similarity measure based on approximations of kl-divergence between two gaussian mixtures," in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, 2003.

[7] Michael D. Garris, "Intelligent system for reading handwriting on forms," in *HICSS '98: Proceedings of the Thirty-First Annual Hawaii International Conference on System Sciences*, 1998.