A Coarse-to-Fine Approach for Handwritten Word Spotting in Large Scale Historical Documents Collection

J. Almazán, D. Fernández, A. Fornés, J. Lladós, E. Valveny
Computer Vision Center – Dept. Ciències de la Computació
Universitat Autònoma de Barcelona
Barcelona, Spain
{*almazan, dfernandez, afornes, josep, ernest*}*@cvc.uab.es*

***Abstract*—In this paper we propose an approach for word spotting in handwritten document images. We state the problem from a focused retrieval perspective, i.e. locating instances of a query word in a large scale dataset of digitized manuscripts. We combine two approaches, namely one based on word segmentation and another one segmentation-free. The first approach uses a hashing strategy to coarsely prune word images that are unlikely to be instances of the query word. This process is fast but has a low precision due to the errors introduced in the segmentation step. The regions containing candidate words are sent to the second process based on a state of the art technique from the visual object detection field. This discriminative model represents the appearance of the query word and computes a similarity score. In this way we propose a coarse-to-fine approach achieving a compromise between efficiency and accuracy. The validation of the model is shown using a collection of old handwritten manuscripts. We appreciate a substantial improvement in terms of precision regarding the previous proposed method with a low computational cost increase.**

***Keywords*-word spotting; historical documents; appearance models; word indexation;**

I. Introduction

There are large collections of handwritten documents in many libraries, museums, and archives. These contain valuable information accessible only by a few scholars. The digitalization of historical handwritten documents (Figure 1) provides access to these large collections of documents to a wider audience and protects them from frequent handling. But information retrieval requires a suitable description of the images' content and full-text search.

Word spotting is a content-based retrieval procedure which results in a ranked list of word images that are similar to a query word image. Thus given a query word image, instances of the same word class are located into the document to be indexed. In the literature, word spotting can be classified in two families: segmentation-based, and segmentation-free approaches.

Segmentation-based approaches [1], [2] require each document image to be segmented at word level, taking advantage of the knowledge of the structure of a document. The main problem of these approaches is that word classification is strongly influenced by over or under-segmentations.

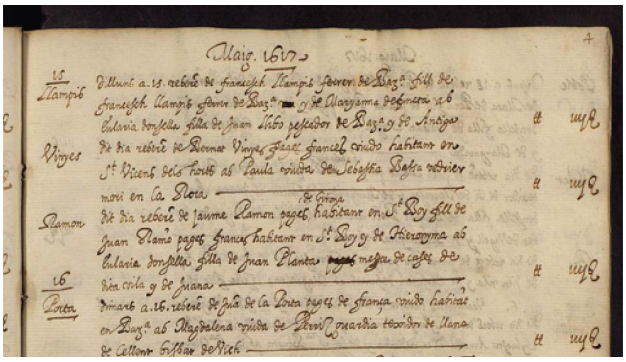

Figure 1. Sample of the *Llibre d'Esposalles* (Archive of Barcelona Cathedral, ACB).

Segmentation-free approaches [3], [4] are not affected by this problem. The image is divided into patches (*e.g.* using a sliding window) and the query word image is classified regarding each patch. Since all the regions are compared, these methods are computationally costly in terms of time.

Related to the segmentation-free approaches, the community of Document Analysis has recently imported ideas form the Computer Vision field. In particular, word spotting is formulated as a subclass of visual object detection [5] modeled by machine learning techniques. However, the use of these state of the art techniques in word spotting has not been commonly exploited. One of the recent works that may be considered is the Deformable Part Model of Felzenszwalb *et al.* [6]. For a given class, they learn different deformable part filters that represent sub-regions of the image, which are described with an Histogram of Oriented Gradients (HOG) [7]. Another method, which is very related, is the work of Malisiewicz *et al.* [8] in the sense that they also learn a filter form HOG descriptor using Support Vector Machine (SVM). The difference lies in that they learn a model for a single image, not for a class given a set of training images. Both methods obtain state of the art results for the difficult task of object detection. However, their direct application to word detection, where we work with large collections of document images, has not been explored.

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.151

When dealing with large collections of handwritten documents, segmentation-based approaches usually cluster the features (segmented words) in bins and construct an inverted file indexation structure to perform word spotting with a focused retrieval scheme. As mentioned before, the performance depends on a good segmentation of words. In this paper we propose a two-level approach that combines the efficiency of indexing strategies and the accuracy of object-detection inspired approaches, such as [6], [8]. The first step is devoted to locate regions of interest in the documents. It is based on a classical approach of word spotting using segmentation and indexation. In this way we discard many regions unlikely to contain instances of the query. Then, the second step is a model-oriented process with the objective of detecting words in the spotted regions obtained in the first step. This second process can be seen as a precise and segmentation-free word detection method whose exhaustive windowing search is only applied in some candidate areas of the documents. So, concretely, our aim is a first efficient method with a high resulting recall – we do not want to filter out areas where the query word is present – and a second method with a high precision to rank all the spotted regions. We try to reach a compromise of efficiency and accuracy.

The novelty of the paper is the combination of a classical word spotting method based on indexation with the most recent techniques in pattern recognition for the detection of visual words in document images. The method is be evaluated in a collection of historical handwritten documents for the task of words detection.

The rest of the paper is organized as follows: Section II is devoted to explain the proposed method. The description of the dataset, the experimental protocols and the performance results, which includes the comparison with different configurations, is presented in Section III. Finally, Section IV concludes the paper.

II. Algorithm Description

Given a database of document images, we first segment all the words. Word images are clustered and organized in a hashing structure in terms of a set of features. In a first spotting stage, a given query image is indexed and candidate word images (locations) are retrieved from the database. This step is designed so it returns many false positives but it guarantees a high recall. The advantage is that it is fast (the index is constructed off-line) and drastically reduces the search space. Afterwards, in a second stage, a finer spotting strategy based on a discriminative model of appearance is performed on the locations retrieved in the first step. Hence, we use this model for each class to find more accurately the positive instances of the queried words class. A visual scheme of the model is shown in Figure 3.

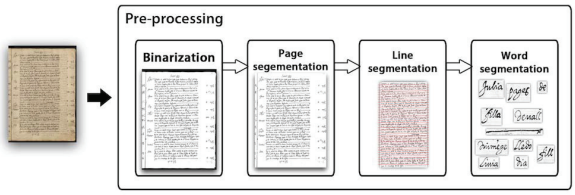

Figure 2. Pre-process: improving the quality of the documents.

A. Word Segmentation

Since Historical documents can be affected by degradations, a preprocessing step is applied before segmenting words. First, the document is binarized, and the margins are removed. The page is then segmented into lines using projection analysis techniques [9]. Once the lines are extracted, words are segmented using a projection function which is smoothed with an Anisotropic Gaussian Filter [1]. The process is shown in Figure 2. For further details, see [10].

B. Word Hashing

This part of the method is based on a previous work [10], which is inspired in characteristic Loci feature [11], [12]. Given a word image, a feature vector based on Loci characteristics is computed at some characteristic points. Loci characteristics encode the frequency of intersection counts for a given key-point in different direction paths starting from this point. As key-points we can use contours, foreground pixels, background pixels or skeletons. In this work we have selected background pixels as key-points experimentally.

Once word images are encoded using a Loci-based descriptor, the indexing structure is organized in a hashing-like way where features are encoded as index keys and words are stored in a hashing structure. Afterwards, the word spotting is performed by a voting process after Loci vectors are extracted from the query word and used as index to access the hashing table. Let us further describe the different steps.

Features are extracted using a pseudo-structural descriptor based on the characteristic Loci features. This feature vector is composed of the number of intersections in the four directions (up, down, right and left). For each background pixel in a binary image, and each direction, the descriptor counts the number of intersections (an intersection means a black/white transition between two consecutive pixels). Hence, each key-point generates a codeword named *Locu number* of length 8. The feature vector is computed by assigning a label to each background (or foreground) pixel. They generate a *Locu number* of length 8 for each keypoint. These values correspond to the counts of intersections with the skeletonized image along 8 directions starting from the

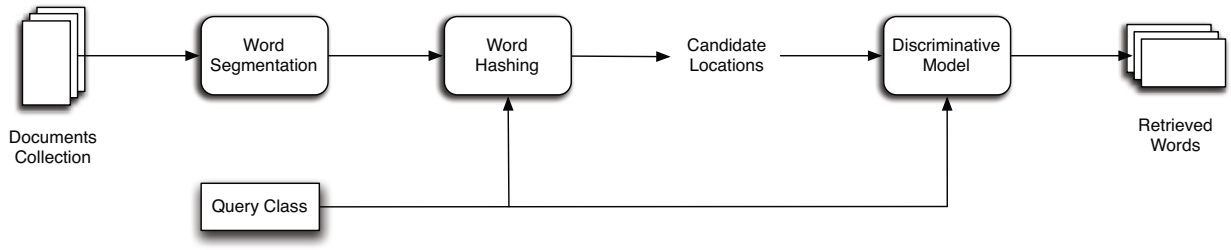

Figure 3. General scheme of the method.

keypoint. A word image is finally encoded with a histogram of *Locu numbers*.

A hash table is created using the obtained *Locu numbers*. For each word of the document the Euclidean distance to the rest of the words is computed.

C. Model of Appearance

Given a set of n training samples of the query word class extracted from the documents, we represent each training image I_i via a rigid HOG template [7], $\mathbf{x_i}$. This descriptor divides the image in equal-sized cells, computes the gradients of the image, and builds a histogram of these gradients in each individual cell. Using a sizing heuristic, we warp all the images to the same size producing the minimum deformation. Then, we create a descriptor for each image with a cell size of 5 pixels. The sizing heuristic attempts to represent each image with approximately 50 cells. So, considering that every cell contains a histogram of length equal to 31[1], the final dimension of the descriptor is $\sim$1,500. This set of images represents the set of positive samples P.

In order to build the discriminative model, we also need a set N of negative samples. It is built in an unsupervised way: a set of sub-images with size equal to the training images is randomly extracted from all the pages. This completely random selection of negative images may cause that a word belonging to the queried class is considered as a negative example. However, if the set of negative examples is large enough, we are still able to learn a representative model of the class [13]. In our experiments we use a negative set N of $10,000$ sub-images. The main advantage of this training method is that a labeled training set is not necessary.

Finally, we learn a model of appearance using a Support Vector Machine (SVM) with sets P and N. The result of the discriminative learning is a set of weights $\mathbf{w}$ on the appearance features (HOG in our case) that provide the best discrimination. We can use these weights to compute an appearance similarity. Given the learned query dependent weight vector $\mathbf{w_q}$, the appearance similarity between the query class and another sub-image I_j can be defined as:

[1]We use the implementation of Felzenszwalb's DPM [6], which introduces some improvements and dimensional reduction over the original implementation of Dalal and Triggs [7].

$$S(q, I_j) = \mathbf{w_q}^T \mathbf{x_j} \quad (1)$$

where $\mathbf{x_j}$ is the feature vector of I_j. We use this appearance similarity as a measure to score sub-images of the document pages and to rank them.

D. Region Spotting and Word Detection

Once we compute the hash structure and the model of appearance, we first spot some regions where the ranking using the model will be performed. So, given the same set P of training images used for training the model of appearance, and using the distance introduced in Section II-B, we select the t closest words to the n training images. This results in a set L of t locations where words belonging to the query class are possibly located. The purpose of this step is to efficiently select interesting regions of the collection of documents and discard big areas of the images unlikely to contain the queried class. Then, we use these regions to select a set of windows from a small area around them. Note therefore that the number of windows will be larger than the number of regions. Using the vector of weights $\mathbf{w_q}$ and Equation 1, we will obtain a score for every analyzed window from all the spotted regions. Finally, we rank all the windows using this score, and we use a standard non-maxima suppression for filtering redundant responses. The resulting windows are the words retrieved by our method.

III. Experimental results

In this section we introduce the dataset and the experiments performed to evaluate our method, and we show the results obtained.

A. Dataset

The experimental dataset is a collection of scanned marriage licenses of the Barcelona Cathedral (*Llibre of Esposalles*). This set of books, written between 1451 and 1905, contains information of every marriage and the corresponding fee paid according to the social status of the families. It is conserved at the Archives of the Barcelona Cathedral and comprises 244 books with information on approximately 550,000 marriages celebrated in over 250 parishes. Each

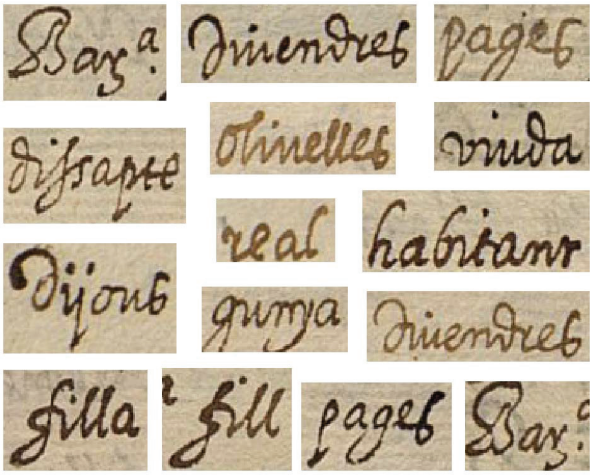

Figure 4. A set of words extracted from the dataset.

book contains the marriages of two years, and was written by a different writer. The original scanned documents have different ink and paper color tonalities, and they are degraded by lifetime and frequent handling. To keep the original appearance, documents were scanned in color. We show an example of the words in Figure 4. Information extraction from these manuscripts is of key relevance for scholars in social sciences to study the demographical changes over five centuries.

The ground-truth that we have used for our experiments consists of 50 pages extracted from a concrete volume. This volume belongs to the same writer. 19 different classes of words have been selected. These word images have been labeled in all the 50 pages.

B. Experiments and Performance Evaluation

We evaluate our method using the above mentioned dataset. For the 19 classes, we tested different values of n training images, which are randomly chosen, repeating the process 10 times. As a retrieval task, we measure the performance of our method using the average precision and the mean recall, giving us a general idea about the position in the ranking of the true positives. An average precision equal to 1 means that all the words have been retrieved in the first positions. Note that the words used for training have not been considered as true positives in the performance measure.

With this experiment configuration, we are going to compare three different configurations of the proposed method:

- **Configuration WH (Section II-B):** We rank the retrieved results using only the **word-to-word** distance similarity based on Loci features. That is, using as final result the spotted locations without applying the model refinement. In this case, in order to achieve a good average precision, we need to set the number of spotted locations t to a certain value where we reach a compromise between precision and accuracy.
- **Configuration WHMA (Section II-D):** We use the complete proposed method: we spot the relevant locations with the **word-to-word** distance similarity, and then we refine the results with the **model of appearance**. Contrary to the previous configuration, we now need in the first step a high recall for a latter refinement, so we can use a higher number of t spotted locations. For our experiments we use $2 \cdot t$.
- **Configuration MA (Section II-C):** We apply the **model of appearance** over the whole document pages without eliminating zones containing words unlikely to belong to the query class.

The comparison, apart from being in terms of average precision (AP) and recall, it is also done regarding the computational time.

C. Results

In Table I we show the results of the three mentioned configurations, **WH**, **WHMA** and **MA** for the word detection task. For this experiment we have set t equal to 4,000. As we can see, the combination of both steps (**WHMA**), outperforms **WH** for all the experiments with different number of training images in terms of mean average precision. The difference in performance increases when the number of training images is higher. Concretely, when using n equal to 10, the mean average precision of **WH** and **WHMA** is 49.38 and 61.70, respectively. Furthermore, if we take a detailed look to the AP of the different classes, we appreciate that it increases for almost all of them, except for the shortest words, *de* and *ab*. This is mainly caused because of an inherent problem in the sliding window technique: short words may be found as sub-words inside longer ones, resulting in a false positive. This may be solved with a additional post-process step. Moreover, we want to point out the increase in AP of our proposed method in classes such as *dilluns*, *dimarts*, *dimecres* and *divendres*. These classes are quite visually similar so the **WH** configuration has some problems to differentiate between them. However, the addition of the model of appearance makes results to improve considerably. It shows us the power of the discriminative model learned, and the capacity of the HOG descriptor to represent document images. Additionally, we see that the configuration **MA** obtains the higher mean average precision. It shows again that the discriminative model of appearance based on HOG obtains a very good performance. However, the high cost to apply it to the whole page in large collections makes unfeasible its application. So, it makes our proposed method, the combination of spotting locations and a model of appearance, a satisfactory compromise between computational cost and performance. Finally, regarding the mean recall, our proposed method results in a 85.09%, which may be increased with a higher

Table I
AVERAGE PRECISION FOR THE DIFFERENT CLASSES USING THE CONFIGURATIONS **WH**, **WHMA** AND **MA**, WITH DIFFERENT NUMBER OF TRAINING SAMPLES (3, 5 AND 10).

Training samples	3		5		10		
Configuration	**WH**	**WHMA**	**WH**	**WHMA**	**WH**	**WHMA**	**MA**
dijous	27.81	34.62	38.49	40.20	25.32	49.37	69.67
reberé	57.50	74.78	68.71	77.51	68.16	75.94	78.38
de	88.77	66.02	88.01	62.89	87.69	69.12	46.11
fill	73.31	42.09	73.89	45.92	72.75	48.15	44.59
ab	83.33	59.42	85.19	66.76	85.16	70.92	57.33
donsella	81.13	75.45	70.74	72.22	83.66	81.52	89.55
filla	44.34	57.39	80.94	58.46	71.41	66.66	73.45
pages	67.37	76.55	79.49	78.61	72.10	80.55	83.40
dia	76.80	46.47	72.62	50.32	67.87	53.08	56.88
habitant	10.87	74.10	49.84	83.57	63.90	87.01	94.67
dit	15.20	48.11	25.22	51.67	29.45	58.78	61.92
viuda	39.25	69.47	66.41	74.00	54.67	77.44	76.55
barna	8.34	40.34	2567	47.71	31.66	48.84	65.56
dissabte	46.81	59.94	49.31	59.20	33.61	67.29	93.97
dilluns	33.24	64.88	8.07	68.22	14.30	71.44	76.04
viudo	34.89	33.13	37.04	32.78	32.56	34.30	61.04
dimarts	24.25	10.51	25.14	7.74	15.51	10.34	25.89
dimecres	0.94	43.25	18.00	46.54	1.45	51.39	68.56
divendres	22.68	65.50	18.34	76.38	26.68	69.55	56.10
mAP	44.04	54.79	51.61	57.95	49.38	61.70	67.33

value of t, but affecting to the computational time efficiency. This affirmation is validated when checking the mean recall of the **MA** configuration, which is 100%.

In Table II, we present the mean average precision and the mean recall obtained for different values of t selected regions using the complete model **WHMA**. Moreover, we include the computational time, which is the average necessary time, in seconds, to perform a search of a query in a single page document. It has been obtained using a prototype version implemented in MATLAB. An optimized implementation in C will drastically reduce the computational time presented. This would be necessary in order to apply this method to real world scenarios. As we can see, the mean average precision increases with the number of spotted locations. In analog way, the behavior of the mean recall is exactly the same. The only problem, as it was expected, is that the computational time also linearly increases, because the number of sub-images to analyze is higher. We can see this value t as a parameter to adjust the trade off between performance and computational time. Finally, note that the total number of regions in a single page is approximately 40,000. So, in order to apply a common sliding window approach – *e.g.* the **MA** configuration – in the complete dataset of 50 document images, we will need to evaluate 2 million regions. If the computational time linearly increases with the number of regions, this makes unfeasible to apply a simple sliding window method. Our coarse-to-fine approach is presented as a feasible solution with a small loss in precision.

In Figure 5 we show some qualitative results. The first row of every query are the first words retrieved by **WH**, and the second row are the words retrieved after the refinement of the model of appearance, that is, **WHMA**. We can appreciate there how the model of appearance re-ranks the results improving the precision. Furthemore, in the case of the query *viuda*, **WH** fails by confusing it with the word *viudo*, but **WHMA** is able to correct it. It shows that the feature descriptor and the discriminative model are able to deal with fine details.

Table II
MEAN AVERAGE PRECISION, MEAN RECALL AND COMPUTATIONAL TIME OF **WHMA** FOR DIFFERENT NUMBERS OF SPOTTED REGIONS. THE TIME IS THE AVERAGE TIME, IN SECONDS, FOR EVERY WORD TO PROCESS A SINGLE PAGE.

regions	2k	4k	6k	8k	10k
mAP	55.92	57.89	58.54	59.74	60.08
mRec	71.06	76.61	80.65	85.09	88.76
sec	0.42	0.85	1.29	1.73	2.22

IV. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a word spotting method using a coarse-to-fine approach. The proposed method combines an efficient indexation method for spotting interesting regions and a precise discriminative model of appearance for ranking this regions and retrieving similar images to the query. In this way we achieve a trade off between computational cost and precision, which is shown in the obtained results. Our methodology outperforms the classical word spotting approaches in terms of average precision with a computational time lower than exhaustive window searching methods. Moreover, we have shown that recent techniques from the Pattern Recognition and Machine Learning fields can be applied to Document Analysis with satisfactory performance.

Figure 5. First words retrieved from configurations **WH** and **WHMA** for queries *divendres* and *viuda*. Red boxes stands for a false positive result.

As future work, we plan to develop a completely unsupervised approach where we do not need to predefine the key words, that is, a word spotting method with an open dictionary. The proposed method can be used in this way when we use a single example to train the model of appearance. The model just needs an unlabeled example, and it retrieves similar images based on appearance. We plan to improve the results using a similar approach to [13], which obtains state of the art performance for image retrieval learning a discriminative model with a single query image. Additionally, and taking advantage of our two-steps coarse-to-fine approach, we can use in a smart way the first region spotting step to retrieve positive and negative examples in order to train a more robust discriminative model. Finally, it is important to note that the method presented is not invariant to scale due to the sliding window approach. A possible solution would be to include the pyramid-HOG [6] where the features are extracted from the images at different scales.

ACKNOWLEDGMENT

The authors thank to the *CED-UAB* and the Cathedral of Barcelona for providing the images. This work has been partially supported by the Spanish projects TIN2011-24631, TIN2009-14633-C03-03 and CSD2007-00018, by the EU project ERC-2010-AdG-20100407-269796 and by two research grants of the UAB (471-01-8/09).

REFERENCES

[1] R. Manmatha and J. Rothfeder, "A scale space approach for automatically segmenting words from historical handwritten documents," *Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1212–1225, aug. 2005.

[2] F. Luthy, T. Varga, and H. Bunke, "Using hidden markov models as a tool for handwritten text line segmentation," in *International Conference on Document Analysis and Recognition*, vol. 1, sep. 2007, p. 8.

[3] Y. Leydier, F. Lebourgeois, and H. Emptoz, "Text search for medieval manuscript images," *Pattern Recognition*, vol. 40, no. 12, pp. 3552–3567, dec. 2007.

[4] B. Gatos and I. Pratikakis, "Segmentation-free word spotting in historical printed documents," in *International Conference on Document Analysis and Recognition*, july 2009, pp. 271–275.

[5] M. Rusiñol, D. Aldavert, R. Toledo, and J. Lladós, "Logo spotting by a bag-of-words approach for document categorization," in *International Conference on Document Analysis and Recognition*, 2011, pp. 63–67.

[6] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramaman, "Object detection with discriminatively trained part based models," *Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, sep. 2010.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[8] T. Malisiewicz, A. Gupta, and A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *International Conference on Computer Vision*, 2011.

[9] V. Papavassiliou, T. Stafylakis, V. Katsouros, and G. Carayannis, "Handwritten document image segmentation into text lines and words," *Pattern Recognition*, vol. 43, no. 1, pp. 369–377, jan. 2010.

[10] D. Fernández, J. Lladós, and A. Fornés, "Handwritten word spotting in old manuscript images using a pseudo-structural descriptor organized in a hash structure," in *Iberian conference on Pattern Recognition and Image Analysis*, 2011, pp. 628–635.

[11] A. Ebrahimi and E. Kabir, "A pictorial dictionary for printed Farsi subwords," *Pattern Recognition Letters*, vol. 29, pp. 656–663, apr. 2008.

[12] H. Glucksman, "Classification of mixed-font alphabets by characteristic loci," *Proc. IEEE Comput. Conf.*, pp. 138–141, sep. 1967.

[13] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. Efros, "Data-driven visual similarity for cross-domain image matching," in *Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH Asia)*, vol. 30, no. 6, 2011.