Affine Invariant Visual Phrases for Object Instance Recognition

Viorica Pătrăucean
University of Cambridge, UK
vp344@cam.ac.uk

Maks Ovsjanikov
LIX, École Polytechnique, France
maks@lix.polytechnique.fr

Abstract

Object instance recognition approaches based on the bag-of-words model are severely affected by the loss of spatial consistency during retrieval. As a result, costly RANSAC verification is needed to ensure geometric consistency between the query and the retrieved images. A common alternative is to inject geometric information directly into the retrieval procedure, by endowing the visual words with additional information. Most of the existing approaches in this category can efficiently handle only restricted classes of geometric transformations, including scale and translation. In this paper, we propose a simple and efficient scheme that can cover the more complex class of full affine transformations. We demonstrate the usefulness of our approach in the case of planar object instance recognition, such as recognition of books, logos, traffic signs, etc.

1 Introduction

Object instance recognition – an effortless task for the human brain [1]– continues to challenge the computer vision community. Given a number of objects with (training) images taken from different viewpoints for each object, the goal is to automatically identify these known objects in new (test) images, without having any prior information on their location, scale, or pose. Numerous research works have tackled this problem over the decades, using geometric invariants [2, 3], geometric hashing [4], or various versions of the bag-of-words (BoW) model [5, 6, 7], to name just a few. In this paper we show how the scalability of the latter category of approaches can be combined with the accuracy gained from informative geometric invariants, in the case of full affine transformations (see Fig. 1).

Methods based on geometric invariants and geometric hashing are powerful spatially-sensitive tools for instance recognition [8, 9]. However, they do not naturally support image indexing schemes for fast retrieval from large datasets, being suitable mostly for one-to-one matching. The BoW model on the other hand, provides a simple and efficient alternative, capable of fast retrieval from large collections. In the original and most basic form of this framework [5, 10], each image is represented as an unordered collection of feature points, which are summarised via a histogram of local image descriptors, quantised to a vocabulary of *visual words*. For efficiency, word occurrences are recorded in an *inverted file*, which associates to each word a *posting list*, storing the labels of the training images that contain that word. Images can then be easily compared through the comparison of the corresponding histograms of visual words. A widely recognized limitation of the original BoW model is its complete insensitivity to the spatial distribution of feature points in the image. An additional geometric verification, e.g. based on RANSAC [11], is needed as a post-processing step to reorder the retrieved candidates, according to the spatial consistency of the matched local features between the test and the training images [12].

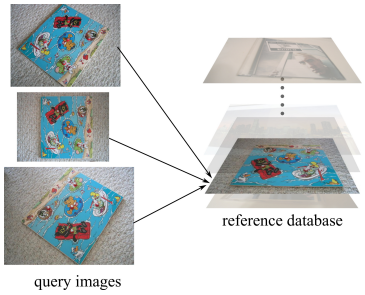

Figure 1: Instance recognition under affine model assumption.

The alternative to the costly RANSAC verification is to inject geometric information directly into the retrieval procedure, by either spatially aggregating the local descriptors in a predefined [6] or adaptively selected [13] set of regions, or by capturing word co-occurrences into *visual phrases*, which correspond to higher-level visual information, either at the level of an entire image [14], or on local neighbourhoods [15, 16]. By attaching additional geometric information to the visual words, schemes that deal with similarity transformations (translation, scale) in the image space have been designed [17, 18]; addressing more complex transformations (e.g. affine) using similar approaches becomes quickly infeasible due to high storage requirements or computational time. To avoid these difficulties, some authors addressed the affine case by reasoning in the feature space: Bronstein and Bronstein build visual phrases using pairs of features [19], whilst Mikulík et al. [7] propose to learn an affine dictionary.

Unfortunately, for the majority of these approaches, the amount of additional spatial information gained is not properly understood, due to the way this information is aggregated in each image [14, 16]. The main contribution of our paper is to propose a simple scheme for obtaining a fully informative representation, by exploiting local descriptors together with precise affine invariants that carry *complete* information necessary for estimating an affine map between candidate image pairs. Remarkably, although 3 feature pairs are necessary to define a unique affine transformation, we show how a quadratic data structure can be used without any loss of information within the BoW model.

Our recognition scheme targets piecewise planar 3D objects, for which the camera projection transformation can be approximated locally by a general affine transformation. We build *affine invariant visual phrases* (AVP) that are able to capture both appearance and geometric information, by combining the

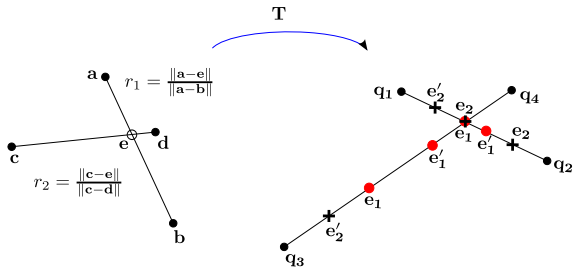

Figure 2: Given a quadruple represented by (r_1, r_2) (left), the potential corresponding intersection points are drawn on each pair (right): $(\mathbf{e}_1, \mathbf{e}'_1)$ are the intersection points generated by r_1 (red dots), and $(\mathbf{e}_2, \mathbf{e}'_2)$ are generated by r_2 (black crosses); the coincidence between a red dot and a black cross indicates an affine transformation $\mathbf{T}$ between the two quadruples.

visual words with geometric invariants in a single, consistent framework. By storing the AVPs in an adapted inverted file, the AVPs that co-occur in the test image and the training images can be efficiently identified at query time, and training images with a sufficient number of such co-occurrences are reported as matching instances.

2 Affine Invariants of Quadruples of Points

Consider the problem of matching two sets of coplanar points $\mathcal{P}$ and $\mathcal{Q}$ that are related by an affine transformation and have full or partial overlap, containing possibly noisy measurements and outliers. Classic RANSAC schemes sample at random a number of minimal sets of point correspondences – three in the affine case – to define transformation candidates, and keep as valid transformation the one that maximises the number of inliers. The overall complexity is $O(|\mathcal{Q}|^3 \log |\mathcal{Q}|)$, where $|\cdot|$ is the cardinality of the point set [20].

The procedure proposed in [21], that we denote *AffineQuadruples*, reduces the complexity by using affine invariants. It samples (non-minimal) quadruple sets of points from $\mathcal{P}$ and extracts affine invariants from them to efficiently identify the possible matching quadruples from $\mathcal{Q}$. Only the transformations computed from such matching quadruples are promising candidates for the true transformation and worth full verification.

A quadruple is defined as a set of four coplanar points $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$, that can be connected by two intersecting line segments (Fig. 2 left). Denote by $\mathbf{e}$ the intersection point. The two ratios defined by the four points and the intersecting point $\mathbf{e}$, $r_1 = \frac{\|\mathbf{a}-\mathbf{e}\|}{\|\mathbf{a}-\mathbf{b}\|}$ and $r_2 = \frac{\|\mathbf{c}-\mathbf{e}\|}{\|\mathbf{c}-\mathbf{d}\|}$, are invariant under affine transformations. Moreover, to check if two quadruples are related by an affine transformation, it is *sufficient* to check if they have equal ratios. Given a couple (r_1, r_2) extracted from a quadruple $(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$ in $\mathcal{P}$, whose intersection point is $\mathbf{e}$ (Fig. 2 left), enumerating all the possible affine transformations between this quadruple and a set $\mathcal{Q}$ containing $|\mathcal{Q}|$ points can be done by considering only *pairs* in $\mathcal{Q}$ in $O(|\mathcal{Q}|^2 \log |\mathcal{Q}|)$ time, and not in $O(|\mathcal{Q}|^4)$ time as expected. Specifically, one draws on each segment joining pairs of points $(\mathbf{q}_1, \mathbf{q}_2)$ in $\mathcal{Q}$, every possible intersection point that could correspond to $\mathbf{e}$ (Fig. 2 right), using the relations:

$$\begin{aligned} \mathbf{e}_1 &= \mathbf{q}_1 + r_1(\mathbf{q}_2 - \mathbf{q}_1) \\ \mathbf{e}_2 &= \mathbf{q}_1 + r_2(\mathbf{q}_2 - \mathbf{q}_1). \end{aligned} \quad (1)$$

Each ratio generates two potential corresponding intersection points as the points are not ordered, i.e. $(\mathbf{q}_1, \mathbf{q}_2)$ can correspond to $(\mathbf{a}, \mathbf{b})$ or to $(\mathbf{b}, \mathbf{a})$. Two quadruples match under an affine model iff one of the intersection points generated by r_1 coincides with one of the intersection points generated by r_2 (Fig. 2 right). By identifying coincident intersection points, the complexity of the affine matching of a pair of point sets can be reduced to $O(|\mathcal{Q}|^2 \log |\mathcal{Q}|)$ time [21].

3 Scalable Affine Invariant Retrieval

The strength of *AffineQuadruples* comes from the fact that given a couple of invariants (r_1, r_2) in $\mathcal{P}$, one can efficiently enumerate all possible matches in $\mathcal{Q}$ by considering only pairs of points. In the more complex scenario of object instance retrieval, one point set (the test image) needs to be matched against an entire collection (the training set). Hence, we need a way to readily access good candidates for (r_1, r_2). Naive approaches like testing exhaustively all (r_1, r_2) that appear in the training set, or indexing using all (r_1, r_2) that appear in the test image, are not suitable since they lead to $O(|\mathcal{Q}|^4)$ *online* complexity either in the number of feature points in the test image or in feature points across all training images, and have low discriminative power. Our scheme combines the strengths of the BoW model and *AffineQuadruples* to counteract these issues: we use the labels provided by the vocabulary of visual words to distinguish points and thus pairs of points, which enables accessing good candidates for (r_1, r_2), allowing us to match quadruples, which in turn vote for the right image.

3.1 Affine Invariant Visual Phrases (AVPs)

Given a dictionary of N visual words $w_{i,i\in\{1\ldots N\}}$, we define an *affine invariant visual phrase* (AVP) as an augmented quadruple, represented as a 6D point $(w_1, w_2, w_3, w_4, r_1, r_2)$: the first four dimensions are the dictionary labels of the four keypoints composing the quadruple, while the last two represent the affine invariant ratios of the quadruple. Each AVP is built so that r_1 and r_2 are associated to the ordered pairs (w_1, w_2) and (w_3, w_4) respectively, with $w_1 < w_2$, and $w_3 < w_4$. Differently from *AffineQuadruples*, in our problem the points are endowed with local descriptor labels, which allows to assign a canonical order. Note that we discard pairs whose keypoints have the same label, e.g. (w_1, w_1), and AVPs in which the two pairs have the same labels, e.g. (w_1, w_2, w_1, w_2), since in these cases, the AVPs cannot be uniquely matched.

3.2 Inverted File for AVPs

Given the set of quantised local descriptors and their locations in each training image, we extract AVPs and store them in an adapted inverted file for retrieval. The feature labels and the ratio of each pair composing the AVPs are stored separately in the inverted file, whose structure is illustrated in Figure 3. Specifically, the

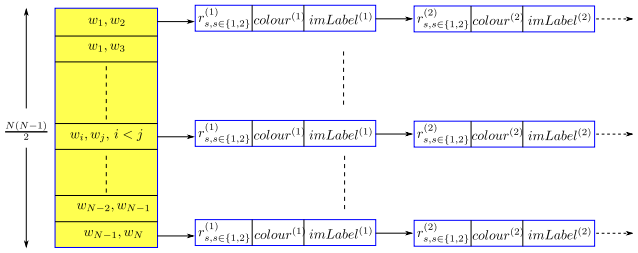

Figure 3: Structure of the inverted file storing AVPs.

inverted file is represented as a list L of posting lists. L contains $\frac{N(N-1)}{2}$ entries, one for each possible ordered pair of feature labels (w_i, w_j), $1 \leq i < j \leq N$. The posting list associated to each pair (w_i, w_j) represents a list of cells which store the labels of the training images in which the pair (w_i, w_j) appeared, together with the ratio associated to this pair in each AVP that contains the pair. An additional field *colour* is stored in each cell, uniquely identifying the AVP, so that at query time, we can identify the two cells generated by the same AVP. We set $colour = b$, for the ratios belonging to the b^{th} AVP. For example, if the b^{th} AVP extracted from training image z is composed of feature labels (w_i, w_j, w_k, w_l), and has ratios (r_1, r_2), then we store the cell (r_1, b, z) in the posting list associated to the pair (w_i, w_j), and the cell (r_2, b, z) in the posting list associated to the pair (w_k, w_l).

To ensure that the size of the inverted file stays within reasonable limits for increasing number of training images, the AVP extraction is not done exhaustively, i.e. we do not consider all possible quadruples from the training images. Instead, for each keypoint f_i we keep its k nearest keypoints, and build all possible AVPs that contain f_i from these $k+1$ points. This is a reasonable choice considering that we deal with 3D objects for which we can hope to have affine invariance only on local (nearly) planar regions of the object. Restricting the AVP extraction to local neighbourhoods does not limit the ability to cope with scale changes, provided the features are stable across scale.

3.3 Voting with AVPs

Given the inverted file described above and the labels of the keypoints detected in the test image, for each pair of keypoints with labels (w_i, w_j) in the test image, we retrieve from the inverted file the posting list associated to the pair. For each cell in the retrieved posting list, we compute the coordinates of the corresponding intersection point $\mathbf{e}$ on the segment joining the pair (w_i, w_j) using one of the equations (1) with the ratio r stored in that cell. The colour of the cell is used as an additional coordinate of the intersection point. In this way, the feature point pairs in the test image that match the same training AVPs will generate intersection points at the same locations in the 3D space (image coordinates, colour). The intersection points are drawn efficiently in $O(n_t^2)$ time, where n_t is the number of keypoints in the test image. The pairs of intersection points having the same colour and located at the same position are identified using a k-d tree and votes are cast for the respective training image. The training images that accumulate high number of votes are kept as matches. To further reduce the execution time, we can restrict the search for pairs to the k nearest neighbours for each keypoint, as done during training for AVP extraction. The key difference with respect to the *AffineQuadruples* setup is that the points, and implicitly the pairs, can be distinguished and ordered due to their dictionary labels. These two qualities reduce unnecessary intersection points, as only the ratios associated to pairs that contain the same labels in the training images are considered. Moreover, since the points in each pair can be ordered by their dictionary label, we need to draw only one intersection point for each ratio, and not two as for *AffineQuadruples*. Semantic-wise, having distinctive and ordinal points makes AVPs strong similarity cues that encode both appearance and geometric information.

3.4 AVPs with Uncertainty

Since the keypoint detection is affected by noise, we need to tolerate a certain error ε when identifying the intersection points generated by the pairs of a matching AVP. The couple of affine invariant ratios used in our scheme is derived from the affine invariant of three collinear points such that $r = \frac{\|a-b\|}{\|a-c\|}$, where (a, b, c) are collinear points. Assuming that the errors produced in keypoint measurements have a standard deviation of $\sigma_a = \sigma_b = \sigma_c$, the standard deviation of the considered invariant can be approximated by [2]: $\sigma_r^2 \approx \left(\frac{\delta r}{\delta a}\right)^2 \sigma_a^2 + \left(\frac{\delta r}{\delta b}\right)^2 \sigma_b^2 + \left(\frac{\delta r}{\delta c}\right)^2 \sigma_c^2$. Scanning the intersection points in the range corresponding to the error $\varepsilon = 2.58\sigma_r$ from the observed r ensures that there is a low probability, approximately 0.01, that a correct corresponding intersection point is missed. The probability of randomly matching affine invariants was analysed in [22]. In our case, due to the fact that the points, and implicitly the pairs, can be distinguished through their dictionary labels, the number of AVPs that are falsely matched decreases for increasing size of the dictionary, as illustrated in Figure 4 left. This result was obtained by running queries with images that do not contain any of the training objects, and by counting the number of (falsely) matched AVPs, while varying the size of the dictionary. As expected, the number of false AVP matches is larger for smaller dictionary sizes, since the keypoint distinctiveness is diminished. Although the risk of having accidental AVP matchings is low, this issue could be further alleviated in future works given that our representation is complete: by storing in the inverted file the positions of the local features in the training images, it is possible to compute the affine transformation associated with each AVP match, and cast a vote in the space defined by *(training image, transformation)* couples.

4 Experiments

We ran recognition tests on a subset of the dataset proposed in [23], which contains numerous planar objects (CDs, paintings, books), with 4 images per object; see example in Fig. 1. We use Hessian-affine keypoints and descriptors to build AVPs [24]. The visual dictionary contains $N = 5000$ words, and it was trained using the k-means clustering algorithm available in OpenCV [25]. We extract AVPs using $k = 30$ nearest neighbours for each keypoint. As a baseline,

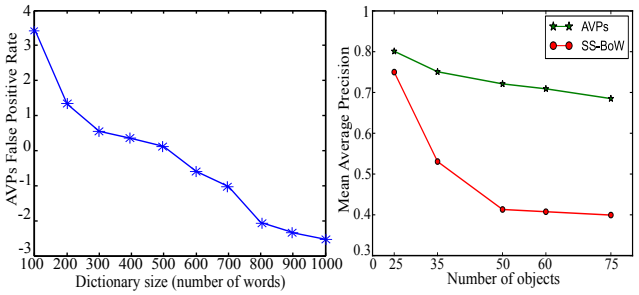

Figure 4: Left: Average number of falsely matched AVPs (log-scale), for increasing dictionary size. Right: Mean average precision for increasing number of objects in the training set.

we consider the method proposed in [19] that we denote by SS-BoW[1] (Spatially Sensitive BoW), which is directly related to our method, addressing the class of affine transformations using visual phrases formed by pairs of keypoints.

The tests consider one of the four images as test image, and the other three as training images, for increasing number of objects in the dataset. The two methods are compared in terms of mean average precision (mAP), with average precision (AP) defined as $AP = \frac{1}{R}\sum_k P(k)$, where R is the number of relevant images in the dataset, and $P(k)$ is the precision at k, i.e. the percentage of relevant images in the top k retrieved images. For this dataset, $R = 3$ and $k \in \{1, 2, 3\}$. mAP is the mean AP over all queries. The results are shown in Fig. 4 right. The proposed method clearly outperforms the baseline, especially for increasing number of training objects, where SS-BoW reports a high number of false matches. This result illustrates the AVP's high discriminant power. The errors reported by our method are caused mainly by the lack of repeatability of the visual words, since the images exhibit significant changes in viewpoint and blur.

5 Conclusion

In this paper, we consider the problem of object instance recognition using the bag-of-words model, and we focus on injecting informative geometric constraints during the initial retrieval. The majority of existing related works are able to efficiently handle only a limited class of similarity transformations. We show that it is possible to cover the more general class of affine transformations using a robust algorithmic scheme with quadratic complexity. The proposed method relies on efficiently storing and matching *affine invariant visual phrases* which encode invariant appearance and affine geometric information.

Acknowledgments

This work was funded by a Google Faculty Research Award, the Marie Curie grant CIG-334283-HRGP, a CNRS chaire d'excellence.

[1] As the code and test data for SS-BoW are not publicly available, we used our own implementation.

References

[1] Riesenhuber, M., Poggio, T.: Models of object recognition. Nature Neuroscience **3** (2000)
[2] Aastrom, K., Morin, L.: Random cross ratios. Technical Report IMAG-RT - 92-088 ; LIFIA - 92-014 (1992)
[3] Meer, P., Lenz, R., Ramakrishna, S.: Efficient invariant representations. IJCV **26** (1998)
[4] Lamdan, Y., Wolfson, H.: Geometric hashing: A general and efficient model-based recognition scheme. In: ICCV. (1988)
[5] Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: ICCV. (2003)
[6] Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR. Volume 2. (2006)
[7] Mikulík, A., Perdoch, M., Chum, O., Matas, J.: Learning a fine vocabulary. In: ECCV. (2010)
[8] Zisserman, A., Forsyth, D., Mundy, J., Rothwell, C., Liu, J., Pillow, N.: 3d object recognition using invariance. Artificial Intelligence **78** (1995)
[9] Reiss, T.H.: Recognizing Planar Objects Using Invariant Image Features. (1993)
[10] Li, F.F., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR. (2005)
[11] Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24** (1981)
[12] Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: CVPR. (2007)
[13] Cao, Y., Wang, C., Li, Z., Zhang, L., Zhang, L.: Spatial-bag-of-features. In: CVPR. (2010)
[14] Torresani, L., Szummer, M., Fitzgibbon, A.: Learning query-dependent prefilters for scalable image retrieval. In: CVPR. (2009)
[15] Zitnick, L., Sun, J., Szeliski, R., Winder, S.: Object instance recognition using triplets of feature symbols. Technical Report MSR-TR-2007-53, Microsoft Research (2007)
[16] Yuan, J., Wu, Y., Yang, M.: Discovery of collocation patterns: from visual words to visual phrases. In: CVPR. (2007)
[17] Jégou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: ECCV. (2008)
[18] Zhang, Y., Jia, Z., Chen, T.: Image retrieval with geometry-preserving visual phrases. In: CVPR. (2011)
[19] Bronstein, A.M., Bronstein, M.M.: Spatially-sensitive affine-invariant image descriptors. In: ECCV. (2010)
[20] Irani, S., Raghavan, P.: Combinatorial and experimental results for randomized point matching algorithms. In: SCG. (1996)
[21] Hopcroft, J., Huttenlocher, D.: On planar point matching under affine transformation. In: First Canadian Conf.Comp. Geom. (1989)
[22] Huttenlocher, D.: Fast affine point matching: an output-sensitive method. In: CVPR. (1991)
[23] Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: CVPR. (2006)
[24] Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. IJCV **60** (2004)
[25] Bradski, G.: Opencv. Dr. Dobb's Journal of Software Tools (2000)