Towards multi-view object class detection

Alexander Thomas, Vittorio Ferrari, Bastian Leibe, Tinne Tuytelaars, Bernt Schiele, Luc van Gool

Towards Multi-View Object Class Detection

Alexander Thomas, KU Leuven (BE)
Vittorio Ferrari, INRIA Grenoble (F)
Bastian Leibe, ETH Zurich (CH)
Tinne Tuytelaars, KU Leuven (BE)
Bernt Schiele, TU Darmstadt (D)
Luc Van Gool, KU Leuven (BE)

Abstract

We present a novel system for generic object class detection. In contrast to most existing systems which focus on a single viewpoint or aspect, our approach can detect object instances from arbitrary viewpoints. This is achieved by combining the Implicit Shape Model for object class detection proposed by Leibe and Schiele with the multi-view specific object recognition system of Ferrari et al.

After learning single-view codebooks, these are interconnected by so-called activation links, obtained through multi-view region tracks across different training views of individual object instances. During recognition, these integrated codebooks work together to determine the location and pose of the object. Experimental results demonstrate the viability of the approach and compare it to a bank of independent single-view detectors.

1. Introduction

Following the development of local, viewpoint adaptive features [12, 13, 20], several powerful systems for the detection of *specific objects* have been proposed recently [7, 11, 18]. These methods are capable of detecting objects in cluttered images in spite of important occlusions and viewpoint changes, given only a few model views. Despite impressive results, these systems suffer from a major limitation: they can only find the specific object shown in the model images.

Another, parallel strand of research using local features has focused on recognizing *classes of objects*, such as cars or airplanes (*e.g.* [2, 3, 6, 9, 16]). Typically, these systems are based on building class-specific clusters of local features with similar appearance which are then treated as object parts and combined spatially in a probabilistic fashion. Such systems have been proven successful, even in cluttered or partially occluded images, and capable of generalizing to previously unseen object instances. However, they are typically limited to a single, predefined viewpoint or aspect (e.g. side-views of cars).

In this paper we integrate two state-of-the-art systems, the multi-view specific object recognition system proposed by Ferrari *et al.* [7, 8] and the Implicit Shape Model for object class detection proposed by Leibe and Schiele [9], so as to combine their complementary strengths. The resulting system is able to recognize novel object instances from arbitrary viewpoints, thereby handling the challenging problem of *multi-view object class detection*.

The Implicit Shape Model [9] (ISM) builds a codebook composed of clusters of local features with similar appearance, and their spatial distribution with respect to the object center across several example instances of a class. During recognition, the test image features are matched to the codebook entries (clusters). Each matched feature activates the spatial distribution associated with a codebook entry and accordingly casts a probabilistic vote for the object position (location and scale). Since votes from features matched to different training images are accumulated, novel objects are detected as puzzles of 'parts' seen in different training objects. However, this scheme requires that all training images be taken from approximately the same viewpoint, and can only detect objects imaged under that viewpoint. A straightforward way to extend it to a multi-view system consists of building a large set of independent single-view detectors and collecting all their output. However, such a naive approach is only effective when there are sufficient single-view detectors to cover all possible viewpoints. Moreover, having many independent detectors leads to a substantial increase in the number of false-positives.

The method proposed in this paper adds to the ISM system a layer of communication between single-view codebooks, which limits the number of views needed to model the object class. This is achieved as follows. The technique proposed by Ferrari *et al.* [8] allows to compute feature tracks densely connecting multiple model views of a single object. We exploit these multi-view tracks to *link* entries across different single-view codebooks. During recognition, these *activation links* are then used to transfer activations between viewpoints. In this fashion, codebooks act *together* and can properly accumulate image evidence coming from features matching to different training views. As a result, intermediate viewpoints are properly covered, which keeps the number of necessary training views low.

In addition to the inter-view activation transfers, we also

introduce a mechanism to select one or a few viewpoints, most likely to match the pose(s) of the object(s) in the test image, and restrict the detection output to them. This avoids the proliferation of false-positives caused by the naive approach of collecting all detections from all training views.

Related Work. Several authors have studied the problem of multi-view object class detection before [17, 19]. Especially in the domain of face detection, dealing with multiple viewpoints (frontal, semi-frontal and profile) is a hot research topic and one of the remaining challenges (*e.g.* [5, 15, 21, 22]). However, most of the proposed methods apply several single-view detectors independently and then combine their responses via some arbitration logic. At best, features are shared among the different single-view detectors to limit the computational overload [19]. In contrast, we no longer rely on single-view detectors working independently, but develop a single integrated multi-view detector that accumulates evidence from different training views at an early stage, thanks to single-view codebooks collaborating by exchanging information via their activation links.

In the context of face detection, some more evolved schemes have been proposed. For instance, [15] studies the trajectories of faces in linear PCA feature space as they rotate, while [22] uses a detector pyramid. For multiview face *recognition*, Fan and Lu [5] integrate feature selection with multi-class classification based on SVM, yielding a discriminative set of features and consequently good recognition results *without* splitting the training data in separate views. However, these methods are specialized for faces and cannot directly be generalized to generic object detection and/or involve human interaction during training.

Also related is the work of Bart *et al.* [1]. They developed a system to recognize specific instances of an object class under arbitrary viewpoint given just a single example view. This is achieved by using so-called *extended fragments*, learnt from short video sequences showing other instances of the same class. Extended fragments are conceptually related to our activation links. Yet, in [1] they are used to link only *two* viewpoints (frontal and 60 degrees), and the application domain (faces) is also different from ours.

The paper is organized as follows. After summarizing the two methods we build upon (sections 2 and 3), we elaborate on the new integrated system in section 4. Section 5 presents experimental results and compares our system to a simple bank of independent single-view detectors.

2. Dense multi-view correspondences by image exploration

The first technique we build upon is the *image exploration* algorithm proposed by Ferrari *et al.* for recognizing specific objects [7] and for establishing dense multi-view correspondences among their model views [8]. In this work, we apply image exploration in the following fashion: for each specific training object, a set of *region-tracks* is produced, densely connecting its model views. Each such track is composed of the image regions of a single physical surface patch along the model views in which it is visible.

Figure 1. *Top: some of the tracks found across 3 views of a motorbike; bottom: all of them.*

In this section we summarize how to obtain the tracks by the method of [7, 8]. First, dense two-view matches are produced between each model image and all other images within a limited neighborhood on the viewing sphere (subsection 2). Next, all pairwise sets of matches are integrated into a single multi-view model (subsection 2).

Dense two-view correspondences. Region correspondences between two model views v_i and v_j are obtained via [7]. The method first generates a large set of low confidence, initial region matches, and then gradually *explores* the surrounding areas, trying to generate more and more matches, increasingly farther from the initial ones. The exploration process exploits the geometric transformations of existing matches to construct correspondences in view v_j, for a number of overlapping circular regions, arranged on a grid completely covering view v_i (*coverage regions*). This is achieved by iteratively alternating expansion phases, which construct new matching regions in v_j, with contraction phases that remove mismatches. With each iteration, the correct matches cover more and more of the object, while the ratio of mismatches progressively decreases. The result is a large set of reliable region correspondences, densely covering the parts of the object visible in both views.

Dense multi-view correspondences. The two-view correspondences, resulting from matching pairs of model views within a limited neighborhood around each view, are now organized into multi-view region tracks [8]. The crucial point is to use always the same coverage regions when matching a certain view to any of the other model views. As a consequence, each region-track is directly defined by a coverage region together with all regions it matches in the other views (figure 1).

3. Object Class Detection with an Implicit Shape Model

When dealing with object categories instead of specific object instances, one has to account for two important conceptual changes. First, one cannot expect to find exact correspondences between test and model views anymore, since only a limited number of examplars of the target category will be available for training. Second, even if similar local structures are found, their spatial location on the object may still vary considerably due to intra-class variation.

Our second basis technique, the *Implicit Shape Model* (ISM) approach proposed by Leibe & Schiele [9], generalizes over object instances by constructing a codebook of local structures that appear repeatedly on the object category. The codebook entries are obtained by clustering image features sampled at interest point locations. Instead of searching for exact correspondences between a novel test image and candidate model views, the ISM approach tries to map sampled image features onto this codebook representation. We refer to the locations in the image that are mapped onto a codebook entry as *occurrences* of that codebook entry. The rigid spatial constraints from object identification approaches are then replaced by spatial occurrence distributions for each codebook entry. Those distributions are estimated by recording all locations a codebook entry could be matched to on the training objects, relative to the annotated object centers. Together with each occurrence, the approach stores a local segmentation mask, which is later used for inferring top-down segmentations.

ISM Recognition. The ISM recognition procedure is formulated as a probabilistic extension of the Hough transform [9]. Let e be a sampled image patch observed at location ℓ. The probability that it matches to codebook entry c_i can be expressed as $p(c_i|e)$. Each matched codebook entry then casts votes for instances of the object category o_n at different locations and scales, *i.e.* $\lambda = (\lambda_x, \lambda_y, \lambda_s)$ according to its spatial occurrence distribution $P(o_n, \lambda|c_i, \ell)$. Thus, the votes are weighted by $P(o_n, \lambda|c_i, \ell)p(c_i|e)$, and the total contribution of a patch to an object hypothesis (o_n, λ) is expressed by the following marginalization:

$$p(o_n, \lambda|e, \ell) = \sum_i P(o_n, \lambda|c_i, \ell)p(c_i|e) \quad (1)$$

The votes are collected in a continuous 3D voting space, and maxima are found using Mean Shift Mode Estimation with a scale-adaptive uniform kernel K [9]:

$$\hat{p}(o_n, \lambda) = \frac{1}{h(\lambda)^3} \sum_k \sum_j p(o_n, \lambda_j|e_k, \ell_k) K(\frac{\lambda - \lambda_j}{h(\lambda)}). \quad (2)$$

For each hypothesis, the ISM approach then computes a probabilistic top-down segmentation in order to determine the hypothesis's support. This is achieved by backprojecting the contributing votes and using the stored local segmentation masks to infer the per-pixel probabilities that the pixel contains *figure* or *ground* given the hypothesis (see [9] for details). Finally, the automatically computed segmentations are used in order to obtain more exact detection scores, taking only *figure* pixels into account, and to disambiguate overlapping hypotheses. This is done in an MDL based verification stage, which searches for the combination of hypotheses that together best explain the image [9, 10].

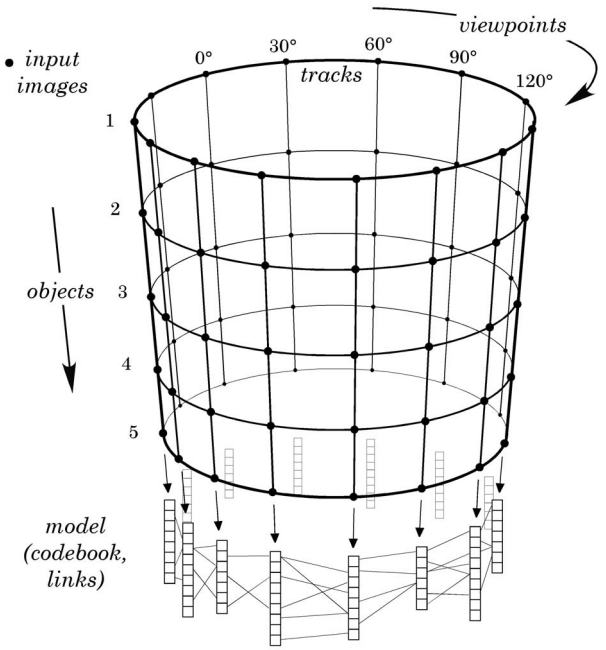

Figure 2. *Visual representation of our multi-view model. Only viewpoints lying on a circle around the object are shown. However, the proposed method supports the general case of viewpoints distributed over the whole viewing sphere.*

4. Integrating the Multi-View Correspondences into the Implicit Shape Model

In this section, we describe how to combine the strengths of the two systems summarized in the previous sections. The goal is to achieve multiview object class detection in a more efficient and more performant way than simply running a bank of single-view detectors. Using *activation links*, learnt from the image exploration algorithm, we can make the different single-view codebooks communicate with each other. This results in additional votes being inserted into a codebook's voting space, based on activations in the other codebooks.

The global scheme of the system is as follows. Initially, both a set of ISM models and exploration systems are trained separately on the same dataset. This dataset consists of images of M object instances, taken from N viewpoints.

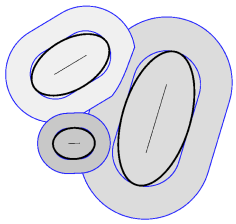

Figure 3. *Attraction zones for regions. The figure shows the areas in which occurrences would be assigned to one of three elliptical regions, using the distance to a line segment as an approximation for the distance to an ellipse.*

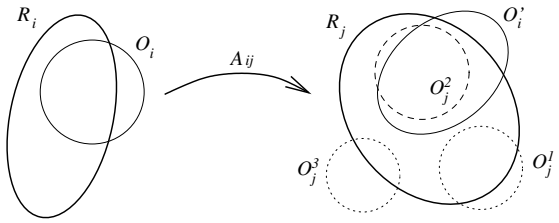

Figure 4. *Establishing links between occurrences. A_{ij} is the affine transformation between the region R_i in view i and R_j in view j. In this example, a link between O_i and O_j^2 is created, because O_j^2 is sufficiently similar to the transformed O_i'.*

The viewpoints should approximately correspond to a fixed set of poses, but each instance does not need to have all viewpoints. In practice, it is sufficient to walk around each of the objects with a camera, and take images at approximately corresponding viewpoints. The total set of training images can be considered as an $M \times N$ matrix, with each row corresponding to an object instance and each column to a viewpoint (figure 2). A set of N ISMs are then trained independently (one ISM for each column), and M sets of region tracks are extracted (one set for each row). The next step is to establish activation links *between* the single-view ISM models. This is explained in section 4.1. Sections 4.2 and 4.3 explain how the multi-view model is used during recognition.

4.1. Establishing Activation Links

The image exploration system (section 2) produces a set of tracks per training object, each containing regions corresponding across the object's model views. These regions are described by ellipses, *i.e.* affine transformations of the unit circle (figure 1). Regions are constructed so that the affine transformation between two regions in a track approximates the affine transformation between the image patches they cover. The goal of the linking stage is to establish connections between the different ISMs. These connections consist of *activation links* between the occurrences, indicating which occurrences in different ISMs correspond to the same object part. Because the ISM and image exploration systems have different goals, they use different features, so there is no one-to-one correspondence between regions and occurrences. Before explaining how to use multi-view tracks to produce activation links, we first report on a subproblem: how to find the region R_i closest to an occurrence O_i. This problem boils down to finding in a set of ellipses (all regions in an image) the one nearest to a point (the center of O_i). An analytical solution for this problem exists, but is computationally expensive. Therefore, we use as an approximation the distance to a line segment of length $\|\mathbf{l}\| - \|\mathbf{s}\|$, aligned with the major axis of the ellipse, with $\mathbf{l}$ and $\mathbf{s}$ the major and minor axes respectively. Occurrences are assigned to the nearest region only if they are within a distance $2 \cdot \|\mathbf{s}\|$. This assumes that the affine transformation of a region is typically valid within a small area around it (figure 3).

With this approximate distance measure, we are now ready to actually link the different ISMs together, by creating activation links between occurrences in different training views. Activation links are created per object instance, *i.e.* they only connect occurrences belonging to a specific training object. The algorithm iterates over all occurrences O_i in all training views of this object. For each O_i, it looks for the nearest region R_i, using the approximate distance measure described above. Then, we treat every other view v_j in the region's track as follows (figure 4). The circular region corresponding to O_i is first transformed with the affine transformation A_{ij} between R_i and R_j, *i.e.* $O_i' = A_{ij} \cdot O_i$. Next, we look for occurrences O_j^k in view v_j that are sufficiently similar to O_i'. All $O_i \rightarrow O_j^k$ are then stored as activation links.

Again, matching the occurrences O_j^k to O_i' involves the comparison between circles and an ellipse. However, this time we do not look for the nearest circle to the ellipse, but for all circles sufficiently similar to the ellipse. We use the following heuristics to determine whether a circle with center $\mathbf{p_c}$ and radius R matches an ellipse with center $\mathbf{p_e}$ and major/minor axis lengths $\|\mathbf{l}\|, \|\mathbf{s}\|$:

$$\|\mathbf{p_c} - \mathbf{p_e}\| < a \cdot R \quad (3)$$

$$\left|1 - (\|\mathbf{s}\| \cdot \|\mathbf{l}\|)/R^2\right| < b \quad (4)$$

$$\|\mathbf{s}\|/R > 1/c \quad (5)$$

$$\|\mathbf{l}\|/R < d \quad (6)$$

with a, b, c, d some parameters, set to $a = 0.35$, $b = 0.25$, $c = d = 3.0$ in all reported experiments. These formulas put constraints on the distance between the centers, the ratio between the areas, the ratio between the minor axis and the radius, and the ratio between the major axis and the radius, respectively.

4.2. Recognition: Selecting Working Views

The early process stages for detecting an instance of the object class in a novel image, are similar to those of the

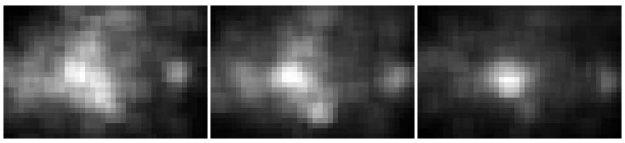

Figure 5. *Voting spaces for three neighbouring viewpoints. Note how strong hypotheses appear at similar locations.*

original ISM framework (section 3). Features are extracted from the image, and matched to all the codebooks of the different ISMs. Next, votes are cast in the Hough spaces of each ISM separately, and initial hypotheses are detected as local density maxima in these spaces. Up to this point, our system works in a similar fashion as a bank of independent single-view detectors.

Then, the first decision our system makes, is which views are likely to match the actual pose(s) of the object(s) in the test image. We will refer to these views as *working views*. A trivial criterion would be to choose those views containing the strongest initial hypotheses, as we expect a strong hypothesis in the correct view. Unfortunately, large amounts of image clutter can also give rise to strong hypotheses. However, we observed that a correct strong hypothesis is most often corroborated by other rather strong hypotheses at similar locations in the voting spaces of neighbouring views (figure 5), whereas this does not hold for the spurious hypotheses caused by image clutter. This can be explained by the fact that there is some continuity in the voting spaces from one viewpoint to the next. Moreover, the pose of an object in a test image mostly falls in between two training views. This phenomenon inspires the following practical algorithm to select working views in a robust way.

We create *clusters* of hypotheses, by picking the strongest hypothesis (across all views), and searching the neighboring views for hypotheses at approximately the same position, *i.e.* within the radius of the adaptive kernel used for density estimation (K in eq. 2). We try to extend the cluster over as many views as possible in all directions, until no more hypotheses are found. Then we continue by taking the next strongest hypothesis that has not yet been assigned to a cluster and repeat the process, until all hypotheses are exhausted. Each cluster is now assigned a score: the sum over the hypothesis scores it contains. Only clusters with score larger than $T \cdot$ (the maximum cluster score) are kept, with T a threshold (0.7 in our experiments). Finally, we select as working views those containing the strongest hypothesis within each remaining cluster.

4.3. Recognition: Transferring Votes Across Views

The next stage is to *augment* the Hough spaces of each selected working view, by inserting additional votes that are stemming from codebook matches in other views. This is where the activation links come into play. Since working

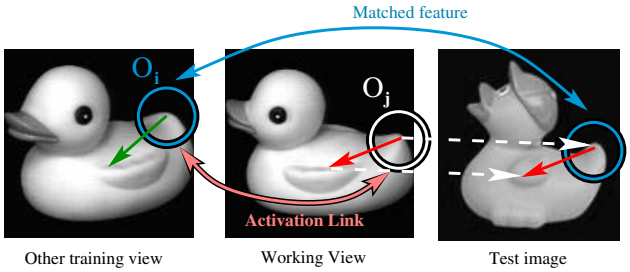

Figure 6. *Vote tranfer. The codebook entry containing occurrence O_i matches to the test image, but another view is selected as working view. Therefore, a vote for O_j is cast.*

views are candidates for the actual pose of the object to be detected, the following process is repeated for each working view. After augmenting the Hough space of a working view, local peaks are detected again, and then the MDL stage of section 3 is performed on the resulting hypotheses. Detections after the MDL stage are the final hypotheses, the output of our system.

The key idea for augmenting the Hough spaces is the following. If a feature matches to a codebook entry in view v_i and there is an activation link between one of the entry's occurrences and occurrences in view v_j, we cast additional votes in view v_j. We call this process *transferring votes*. In other words, if we detect a part in the codebook of view v_i, but we have found view v_j to be a more likely pose for the object, we transfer the evidence of the part to view v_j. Therefore, to cast the transferred vote we use information from both view v_i's and v_j's ISMs. Remember that during the original voting stage, votes are cast to positions computed as the sum of the position where a codebook entry matches in the test image, and the relative positions of the occurrences to the center of the object in the training images. To determine the position of a transferred vote, we assume that when detecting a part in view v_i, the same part may be present in view v_j at approximately the same position. Therefore, the position of the transferred vote is calculated as the sum of the coordinates where the codebook entry matched in view v_i, and the relative coordinates of the occurrence in view v_j (figure 6). Since the estimate for the object center is inevitably less accurate than in the single-view case, we use a larger kernel size when detecting peaks in the augmented Hough spaces. This compensates for the larger variance in the votes' positions.

The weight of the transferred votes is determined by extending eq. 1 to the multi-view system. This formula expresses the contribution of a patch e to an object hypothesis (o_n, λ):

$$p(o_n, \lambda | e, \ell) = \sum_k P(o_n, \lambda | c_k^j, \ell) p(c_k^j | e) + \sum_k \sum_l P(o_n, \lambda | c_k^j, c_l^i, \ell) p(c_l^i | e) \quad (7)$$

Figure 7. *A few test images of the PASCAL VOC challenge (left) and our sports shoe test set (right).*

Figure 8. *A few training images for the shoe model.*

with v_j the current working view. The first term is as in eq. 1. The summation over k runs over all codebook entries for view v_j. The summation over l runs over all other codebooks' entries, *i.e.* for views $v_i \neq v_j$. In this summation, the factor $p(c_l^i|e)$ is the probability that entry c_l^i is a correct interpretation for patch e. Just like in the original ISM system, we assume a uniform distribution here. $P(o_n, \lambda|c_k^j, c_l^i, \ell)$ is non-zero only if there exists an activation link between c_l^i and c_k^j. It expresses the spatial distribution of transferred votes from occurrences in codebook entry c_l^i to occurrences in codebook entry c_k^j. This distribution consists of a set of weighted Dirac-impulses in the 3D Hough space at locations as described above. The weights of these impulses are derived as follows. Each of the K occurrences in codebook entry c_l^i has probability $1/K$ to yield the correct vote for the object center (under the uniform distribution assumption). If this occurrence has L links towards view v_j, the probability for each link to be valid is $1/L$. Therefore, each impulse in the transferred vote distribution should be weighted by $1/(KL)$. Note that, compared to the weights of the *direct* votes, which originate from view v_j itself, there is an additional factor of $1/L$. The weights of transferred votes are lower than direct ones, which adequately mirrors the fact that they are more numerous and less reliable (individually).

5. Results and Conclusions

We report results on two classes: motorbikes and sports shoes. For the motorbikes, a large benchmark test set with images of motorcycles in *various poses* is publicly available from the PASCAL Visual Object Classes (VOC) Challenge [4]. More precisely, we use the 'motorbikes-test2' set, which contains a total of 179 images[1]. As can be seen in figure 7, this is a very challenging test set, due to the large variability in the scale and poses at which the motorcycles appear, the extensive clutter, the often low image quality, and poor imaging conditions. This is confirmed by the only modest results obtained on this set by the various participants in the PASCAL VOC challenge [4, pp. 58] [2] .

Our training set consists of photographs of 30 motorbikes (figure 1). Each training image is segmented so as to roughly isolate the motorbike, which is a requirement for training the ISM models [9]. We underline that this is only necessary for training, and that the test images are input in the system as they are. We use 16 training views, taken every approximately 22.5° on a circle around the object. For practical reasons, it was impossible to collect all 16 viewpoints for all 30 training motorbikes. In fact, only 3 bikes have all 16 images. On average, a motorbike is imaged from 11 views. As a result, for each viewpoint there are an average of 22 object instances, which is only a small number to train the ISMs. Typically about 100 instances are used by Leibe *et al.* [9], and 217 were given to participants of the PASCAL challenge for this dataset. Hence, we expect performance to rise beyond the levels we report here, once more training bikes are included.

As second class we selected sports shoes. The training set (figure 8) contains 16 views around each object, taken at 2 different elevations (i.e. 8 views per elevation). Because there is no standard test set available for this class, we constructed our own set in a similar fashion as in the VOC Challenge. A total of 101 images were collected from Google, Flickr, and Fotolog.com using search terms such as 'sport shoe' or 'trainers'.

In order to evaluate the benefits brought by our multi-view method, we compare it to a bank of the same 16 ISM models, but without any inter-view communication. In this baseline system, each of the 16 ISMs is run separately on the test images, with all their detections being collected and output together. In case two or more detection bounding-boxes overlap more than 40%, we only keep the strongest one (in terms of MDL score). This filter is applied for both the multi-view system and the bank of detectors. We adopt the same evaluation protocol as the PASCAL Challenge: a detection is counted as correct if its bounding-box overlaps more than 50% with the ground-truth one, and vice versa.

Figure 9 shows precision/recall curves for both the bank of detectors and the multi-view system on the motorbike set, when using DoG interest points and simple image patches as descriptors, just as in the original ISM work [9]. Our system's curve shows a substantial improvement in precision compared to the bank of detectors. Besides, unlike the bank of detectors, our method does not need to perform the full ISM recognition for all views. As a result, it offers a 2 to 3 times speed-up. Finally, we also evaluated the influence of the initial feature set, by replacing the previous DoG+Patches set by Hessian-Laplace interest points and Shape Context features, which have yielded superior performance in previous evaluations [14]. As can be seen in figure 9 the new features improve performance

[1] After removing all duplicates from the 202 images mentioned in [4]

[2] www.pascal-network.org/challenges/VOC

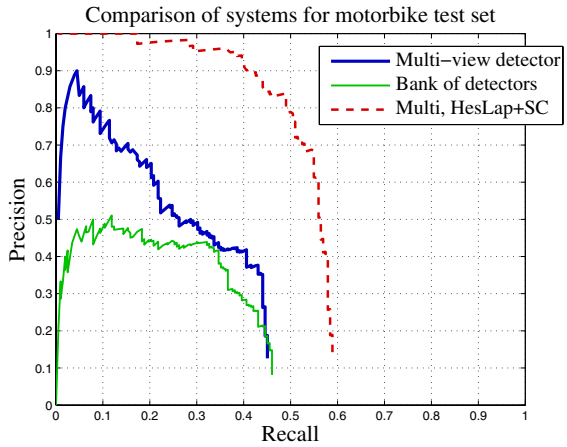

Figure 9. *Precision-recall curves for the motorbike experiments.*

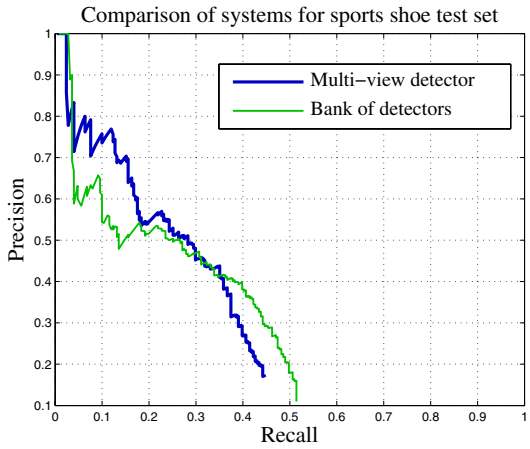

Figure 10. *Precision-recall curves for the sports shoe experiments.*

even further. In future work, we will evaluate how this affects the performance/speed trade-off with the bank of detectors. Compared to the competitors in the PASCAL VOC challenge [4, pp. 58], we rate second with DoG+Patches and outperform all other approaches with the new HesLap+SC features. Considering that we trained our ISMs from much fewer motorbike instances than the participants in the challenge, our system achieves a remarkable performance. However, a perfect comparison is not possible as we trained on different instances and used multiple training views per instance. Figure 11 shows a few examples where the multi-view system correctly detected the motorbike, whereas the bank of detectors failed. These results confirm both main advantages brought by the proposed system. Of course, several cases are not detected by our multi-view system either, while the bounding boxes of others are not estimated to a sufficient accuracy to be counted as correct detections (figure 13). Missing detections are often due to the motorbike being too different from any in our small set of training instances.

Last but not least, figure 10 shows the performance curve for the shoe experiment, while example detections are reported in the bottom row of figure 12. Following our observations on the motorbikes, this experiment was carried out directly with the HesLap+SC features. Although the absolute performance level is lower than for the motorbikes, we regard the performance of our system as satisfactory, given the superior difficulty of this test set. Once again, our system performs better than a bank of independent detectors.

Figure 11. *Left: results of the multi-view system; right: results of the bank of detectors.*

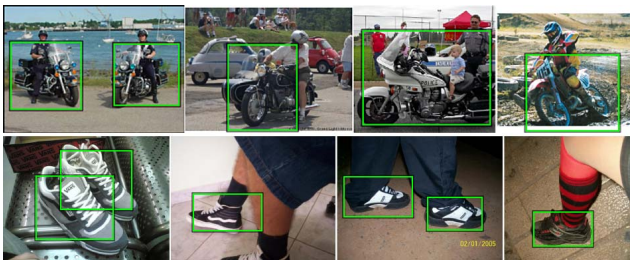

Figure 12. *Some more correct detections, for motorbikes (row 1) and for sports shoes (row 2).*

Discussion. Thanks to the transfer of votes between views, a local object part can vary its pose, relative to the entire object, and still contribute to the detection, as long as it stays approximately at the same relative location. A good case in point is the front wheel of a motorbike, which might be turned in different ways. As such, it will be matched to some view other than the working view (corresponding to the global object's pose). Nevertheless, thanks to vote transfers, it contributes to the correct object position hypothesis in the working view.

The proposed vote transfer scheme can improve recognition performance also in other cases, such as when lighting

Figure 13. *A few missed detections on motorbikes and shoes, due to a too large difference with training instances, or poor contrast.*

conditions vary, or when the object shape changes significantly across different object instances, causing local object parts to be more similar to their counterparts in a different view. Besides, the pose of the object in the test image may fall in-between two training views. In such a case, matches will naturally scatter over both views. By transferring votes, evidence from both views is properly accumulated. Finally, the vote transfer mechanism also relaxes the constraints on the training images. Indeed, the images of different object instances need not be taken from exactly the same poses, as the system is able to transfer information between the codebooks of each pose. On the contrary, slight variations in pose within the same codebook can increase the robustness and the ability to interpolate between views.

The selection of working views further increases the performance of our system. Indeed, by suppressing other views, the number of false positives is reduced. For example, all three single-view detectors would yield a detection for the case in figure 5. However, only one has the correct pose. By selecting a working view, only a single, correct detection is output by the system.

Conclusion. We have proposed a novel system for multi-view object class detection. It integrates two state-of-the-art systems, the image exploration method of Ferrari *et al.*, and the Implicit Shape Model of Leibe and Schiele. The key contributions are the introduction of activation links for transfering votes across views, and of a robust mechanism for working view selection. As experiments show, our approach outperforms a bank of independent single-view detectors.

Future work includes: 1) extend the MDL verification stage to integrate output from different working views, 2) remove the need for the training views to be ordered according to viewpoint, 3) experiment with more classes.

Acknowledgements. The authors gratefully acknowledge support by IWT-Flanders, the European project CLASS, the EU NoE Pascal, and FWO-Flanders.

References

[1] E. Bart, E. Byvatov, S. Ullman, View-invariant recognition using corresponding object fragments", *ECCV*, vol.2, pp. 152-165, 2004.

[2] C. Dance, J. Willamowski, L. Fan, C. Bray, G. Csurka Visual categorization with bags of keypoints International Workshop on Statistical Learning in Computer Vision, 2004

[3] G. Dorko, and C. Schmid, Selection of Scale-Invariant Parts for Object Class Recognition, *ICCV*, 2003.

[4] M. Everingham *et al.*, The 2005 PASCAL Visual Object Class Challenge, Selected Proceedings of the 1st PASCAL Challenges Workshop, to appear

[5] Z.-G. Fan and B.-L. Lu. Fast Recognition of Multi-view Faces with Feature Selection *ICCV*, 2005.

[6] R. Fergus, P. Perona, and A. Zisserman, Object Class Recognition by Unsupervised Scale-Invariant Learning, *CVPR*, 2003.

[7] V. Ferrari, T. Tuytelaars, and L. van Gool, Simultaneous Object Recognition and Segmentation by Image Exploration, *ECCV*, 2004.

[8] V. Ferrari, T. Tuytelaars, and L. Van Gool, Integrating Multiple Model Views for Object Recognition, *CVPR*, Vol. II, pp. 105-112, 2004.

[9] B. Leibe and B. Schiele. Scale-Invariant Object Categorization using a Scale-Adaptive Mean-Shift Search, *DAGM*, pp. 145-153, 2004.

[10] B. Leibe, E. Seemann, and B. Schiele. Pedestrian Detection in Crowded Scenes *CVPR*, 2005.

[11] D. Lowe, Local Feature View Clustering for 3D Object Recognition, *CVPR*, 2001.

[12] J. Matas, O. Chum, M. Urban and T. Pajdla Robust Wide Baseline Stereo from Maximally Stable Extremal Regions, *British Machine Vision Conf.*, pp. 414-431, 2002.

[13] K.Mikolajczyk and C.Schmid An affine invariant interest point detector, *ECCV*, vol. 1, pp.128–142, 2002.

[14] K.Mikolajczyk and C.Schmid A performance evaluation of local descriptors, *PAMI*, vol. 27(10), 2005.

[15] J. Ng and S. Gong Multi-view face detection and pose estimation using a composite support vector machine across the view sphere, Proc. Int. Workshop Recognition, Analysis and Tracking of Faces and Gestures in Real Time Systems,pp.14, 1999.

[16] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, Weak Hypotheses and Boosting for Generic Object Detection and Recognition, *ECCV*, pp.71-84, 2004.

[17] H. Schneiderman and T. Kanade, A Statistical Method for 3D Object Detection Applied to Faces and Cars, *CVPR*, vol. 1, pp. 1746, 2000.

[18] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce, 3D Object Modeling and Recognition Using Affine-Invariant Patches and Multi-View Spatial Constraints, *CVPR*, 2003.

[19] A. Torralba, K. Murphy, and W.T. Freeman, Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection, *CVPR*, vol. 2, pp.762-769 , 2004.

[20] T. Tuytelaars and L. Van Gool Wide Baseline Stereo based on Local, Affinely invariant Regions, *British Machine Vision Conference*, pp. 412-422, 2000.

[21] M. Weber, W. Einhaeuser, M. Welling and P. Perona Viewpoint-Invariant Learning and Detection of Human Heads, *Proc. 4th Int. Conf. Autom. Face and Gesture Rec.,* FG Grenoble, France, 2000-3.

[22] S.Z. Li and Z. Zhang, FloatBoost Learning and Statistical Face Detection *PAMI*, 26(9):1112-1123, 2004.