What is the Spatial Extent of an Object?

J.R.R. Uijlings[1] A.W.M. Smeulders[1] R.J.H. Scha[2]

[1]Intelligent Systems Lab Amsterdam,
Informatics Institute,
University of Amsterdam.
{jrr.uijlings,ArnoldSmeulders}@uva.nl

[2]Institute for Language, Logic
and Computation,
University of Amsterdam.
scha@uva.nl

Abstract

This paper discusses the question: Can we improve the recognition of objects by using their spatial context? We start from Bag-of-Words models and use the Pascal 2007 dataset. We use the rough object bounding boxes that come with this dataset to investigate the fundamental gain context can bring. Our main contributions are: (I) The result of Zhang et al. in CVPR07 that context is superfluous derived from the Pascal 2005 data set of 4 classes does not generalize to this dataset. For our larger and more realistic dataset context is important indeed. (II) Using the rough bounding box to limit or extend the scope of an object during both training and testing, we find that the spatial extent of an object is determined by its category: (a) well-defined, rigid objects have the object itself as the preferred spatial extent. (b) Non-rigid objects have an unbounded spatial extent: all spatial extents produce equally good results. (c) Objects primarily categorised based on their function have the whole image as their spatial extent. Finally, (III) using the rough bounding box to treat object and context separately, we find that the upper bound of improvement is 26% *(*12% *absolute) in terms of Mean Average Precision, and this bound is likely to be higher if the localisation is done using segmentation. It is concluded that object localisation, if done sufficiently precise, helps considerably in the recognition of objects for the Pascal 2007 dataset.*

1. Introduction

The popular Bag of Words framework produced the best results in two image/video retrieval tasks of 2008: The Pascal VOC challenge [4] and the TrecVid Video Retrieval task [18]. Its success sparks research on various aspects of the Bag of Words framework. These aspects include point detectors [12], point or patch descriptors [12, 22], visual vocabulary [14, 13, 22], learning methods [22], and inducing spatial information [9]. In this paper we will focus on another aspect within this framework, namely the spatial context of an object.

The role of context within the Bag of Words framework was addressed earlier by Zhang *et al*. [22] as part of their comprehensive comparison of features and kernels. Their experiments on context were performed on the four-class problem of the Pascal VOC 2005 object verification challenge. They report that ”while the backgrounds [context] in most available datasets have non-negligible correlations with the foreground objects, using both foreground [object] and background [context] features for learning and recognition does not result in better performance for our method.” This result was surprising given the importance of context in both psychological work ([1, 2]) and work in computer vision ([5, 16, 17, 19]). Indeed, in the first part of our paper we repeat and extend the experiments from [22] on the larger and more realistic Pascal 2007 dataset and we arrive at a different overall conclusion.

Therefore this paper revisits the role of context in a Bag of Words approach, addressing the following questions: (I) Is context really superfluous? (II) What is the spatial extent of an object? *I.e*. what is the ideal window to look at an object in a Bag of Word fashion? (III) What is the potential performance gain by dividing context from object patches through rough localisation?

2. Related Work

Within the Bag of Words framework Nowak *et al*. [15] and Jurie and Triggs [7] showed that sampling many different patches using either a random strategy [15] or a regular dense grid [7] works better than using interest points (as used for example in [22]). We will adopt the Dense Sampling method of [7].

Mikolajczyk and Schmid [12] and Zhang et al. [22] found that SIFT or SIFT-like variants are the best performing patch descriptors within the Bag of Words framework, so we will use a SIFT-variant.

Large codebooks obtained by unsupervised clustering such as k-means give good performance (*e.g.* [22]). More recent interesting research is focusing on creating very large codebooks using tree-based methods to keep computational demands within bounds [14, 13]. However, as its value for large datasets with a large number of classes has yet to be established, we stick to k-means in this paper.

Support Vector Machines (SVMs) are the most popular classifier in object verification due to its robustness against large feature vectors and sparse data. The choice of SVM kernel has a large influence on performance. Both Zhang *et al*. [22] and Jiang *et al*. [6] concluded that χ^2 is the preferred kernel. We follow their conclusion.

The original Bag of Words framework is orderless. Therefore Lazebnik *et al*. [9] introduced a weak spatial order through their spatial pyramid, in which an image is increasingly divided and codebook frequency histograms are obtained from each region separately. A substantial increase in performance was obtained. However, the spatial pyramid is incompatible with other divisions of the image: all possible regions need to exist to calculate proper codebook frequency histograms for an image. This is not guaranteed when using multiple divisions, hence we have refrained from using the spatial pyramid here.

In Tuytelaars and Schmid [20] it was found that the most characteristic SIFT-patch for sky measures the horizon. Our research examines in general the contribution of non-object patches for object recognition.

Chum and Zissermann [3] and Lampert *et al*. [8] use Bag of Words methods for object *localisation*. In their research they focus on the object. Our research investigates the inclusion of context as an extra information channel. Our research suggests which object classes in such methods are likely to be found and which are not.

3. Experimental Setup

Our paper addresses the role of contextual and object patches in a Bag of Words framework. We do this using a theoretical setting: the Pascal 2007 dataset provides ground truth annotation in the form of bounding boxes around the object which we will use in both the train- and test-phase. Either we use `all` patches from the image without any differentiation or we employ the bounding boxes to distinguish between two types of patches: `object` and `context` patches. Patches without any overlap with the bounding boxes are `context`-patches. These measure context only. Patches that have overlap with the bounding box are `object` patches. Thus `object` patches include pure object patches and object-context *transition* patches, but also include patches measuring only context; these are typically patches at the corners of the bounding box.

We perform the following experiments:

I What is the information content of context?

We repeat and extend parts of the experiments of Zhang *et al*. [22]. We do four runs: (1) A baseline experiment using `all` patches for both training and testing. (2) We train on `all` patches and test on `object` patches. (3) We train and test on `context` patches. (4) We train and test on `object` patches.

II Can context be used as an extra information channel?

We train and test while using `object` and `context` patches as separate information channels.

III What is the spatial extent of an object?

We shrink and enlarge the bounding boxes to determine the best size for recognising objects in a Bag of Words framework.

IV What is the Influence of Object Localisation Accuracy?

We add a random localisation error to the bounding boxes during the test phase to determine its influence on recognition performance.

3.1. Dataset

All experiments are done on the Pascal VOC 2007 challenge. This dataset consists of 9963 images from `www.flickr.com`. It contains twenty different object classes (see figure 1a) and some images contain multiple classes. The dataset is split into two predefined train and test sets of size 5011 and 4952 images respectively.

It should be noted that for all classes except *horse* the photographs have a reasonable variability. The *horse* class is severely biased as it is dominated by two types: images of one specific horse jumping contest, and images of the same photographer taken in the same area and whose name is on the image in white letters. In effect, the horse class shows all signs of an object against a fixed background.

Given a target object, the goal is to generate a ranked list. Performance is measured by evaluating this ranked list using the Average Precision measure, defined as

$$\frac{1}{m}\sum_{i=1}^{n}\frac{f_c(x_i)}{i}, \tag{1}$$

where: n is the number of images. m is the number of images of class c. x_i is the i-th image in the ranked list $X = \{x_1, \cdots, x_n\}$. Finally, f_c is a function which returns the number of images of class c in the first i images if x_i is of class c and 0 otherwise. This measure has range $(0, 1]$ where a higher number means better performance.

3.2. Bag of Words Implementation Details

In our Bag of Words experiments we use a variant of SIFT [10]. Normally, SIFT divides a patch into 4 by 4 subpatches where for each sub-patch a Histogram of Oriented

Gradients is calculated. As we noted that a 2 by 2 SIFT performs marginally better but never worse than the 4 by 4 SIFT when employing the same set of pixel values, we prefer to use the 2 by 2 SIFT as it is computationally much more efficient. The results will generalize to the common 4 by 4 SIFT (data not shown).

Our version of the Dense Sampling strategy [7] samples patches of 8 by 8 pixels at every 4-th pixel. This generates about 8000 patches or SIFT-features per image. Sampling at multiple scales generally gives slightly better results, but preliminary results reveals only small performance increases for all classes. They have no influence on the observations in this paper. The advantage of small patches is that there are less ambiguous patches in terms of `object` or `context` patches. Using a single scale is computationally more efficient.

We obtain a visual vocabulary of 4096 visual words using k-means clustering. The support vector machine (SVM) with a χ^2 kernel [22] is used for learning. Parameter tuning for the SVM is done on the training set using cross-validation.

The above results in a state-of-the-art Bag of Words pipeline [11, 21].

3.3. Evaluation Matrix

We developed a novel way of visualising the data. Instead of using only the Average Precision value per class, we propose to use a confusion matrix based on this measure, which we call Confusion Average Precision Matrix or CAMP. The CAMP includes the Average Precision in its diagonal elements, and provides useful extra information as can be seen by looking at figure 1a. Hence the CAMP allows for a more detailed analysis.

We calculate the CAMP as follows. Whenever a non-target class is encountered at position i, we calculate the difference between the Average Precision score assuming a perfect ranking from position i and a perfect ranking from position $i+1$ where i is a non-target class sample. This difference measures the loss L incurred by having a non-target class at position i. If we define $\hat{f}_c$ as a function which returns the number of images of class c in the first i images, and let $r = m - \hat{f}_c(x_i)$, we can calculate the loss L for the Average Precision measure as

$$L(x_i) = \frac{1}{m}\left(\sum_{j=1}^{r}\frac{\hat{f}_c(x_i)+j}{i+j-1} - \sum_{j=1}^{r}\frac{\hat{f}_c(x_i)+j}{i+j}\right). \quad (2)$$

The confusion with a non-target class is simply the sum of losses of all images containing this non-target class. When an image contains different object classes, the loss is divided over these classes. All confusion scores plus the Average Precision for a single class add to one.

4. Results

4.1. The Information Content of Context

We repeat the experiments of Zhang *et al*. [22] on the Pascal 2007 dataset and extend them by employing Confusion Average Precision Matrices (CAMPs).

4.1.1 Baseline: Use All Patches

For our baseline experiment we use `all` patches in the Bag of Words system as described above. The Mean Average Precision over all classes is 0.46, which is comparable with the results by van de Sande *et al*. [21] and Marszalek *et al*. [11] when using intensity based SIFT with a single-scale Dense Sampling strategy.

The Confusion Average Precision Matrix (CAMP) is given in figure 1a. The rows in the CAMP represent the target class, the columns represent the classes that were found. The classes used in [22] are marked with an asterisk (*).

The bias towards classes with a high number of samples inherent to the Average Precision measure causes the *person* column to be relatively high. Even so, *persons* are confused as *bottles* much more than random. Our subsequent experiments suggest why.

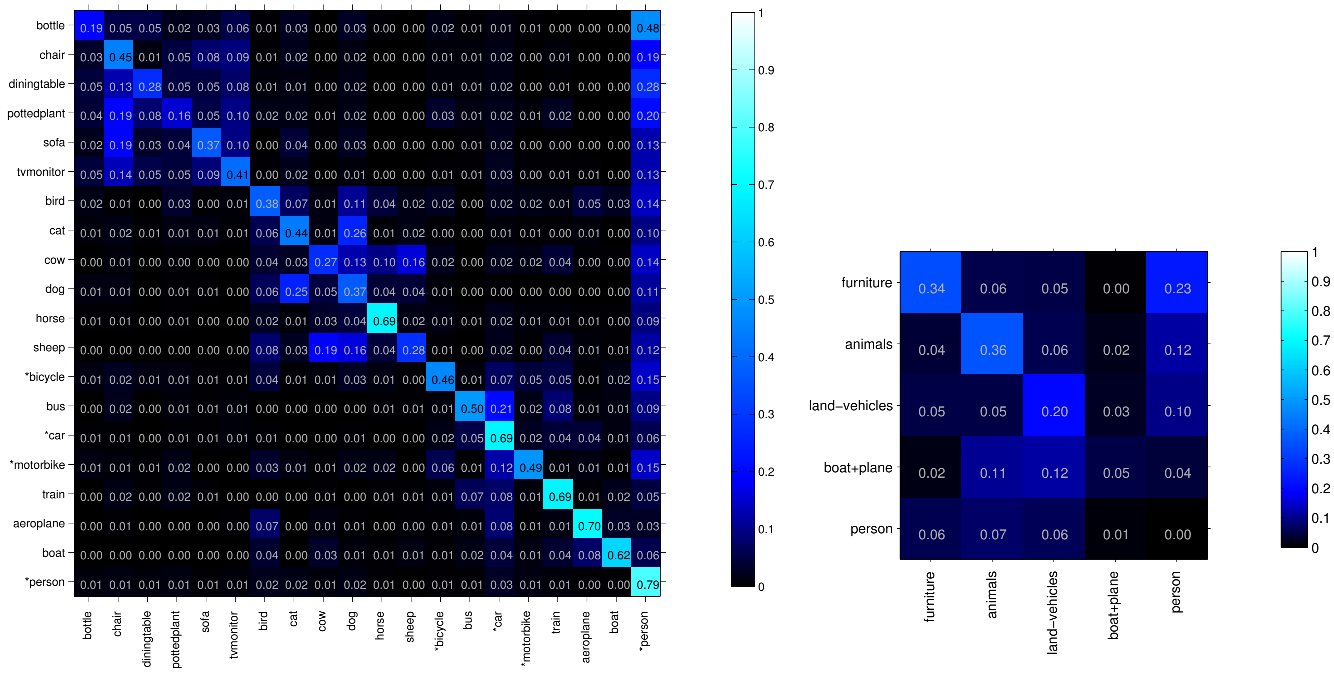
According to the behaviour of the classes, we can roughly divide them into five clusters where most of the confusion concentrates: furniture, animals, land-vehicles, boat+plane, and person. The CAMP when using clusters is presented in figure 1b. This matrix only contains confusion, *i.e*. the off-diagonal elements of figure 1a. The identification of these clusters facilitates subsequent discussion.

4.1.2 Learn All, Test Object Patches

In this experiment we learn by using `all` patches and test on the `object` patches. The resulting confusion matrix looks similar to figure 1a. Therefore we provide the difference between the confusion matrix of this experiment and the baseline experiment in figure 2a. We will use such difference matrices for the subsequent experiments in this section.

We make the following observations. In accordance with Zhang *et al*. [22], *bicycle, car, motorbike,* and *person* either increase in performance or are unaffected. However, for most other classes there is a significant decrease in performance. In fact, the Mean Average Precision over all classes goes down by 0.05, showing that context is learned in this dataset.

For the furniture cluster the context helps with distinguishing them from land-vehicles and persons, as can be seen by the increase in confusion with these clusters (in light-yellow). However, the shared furniture context also causes confusion, as can be seen from the decline in in confusion between furniture classes (in dark-blue).



(a) Class CAMP Baseline

(b) Cluster CAMP Baseline

Figure 1: Confusion Average Precision Matrices (CAMPs) where rows denote the search class and columns denote class-items retrieved, specified in terms of Average Precision. In the CAMP of clusters only the confusion, *i.e.* the off-diagonal elements, is presented. This is the baseline experiment using `all` patches.

Without context a car is less confused as a bus. But a bus is not less confused as a car. This suggests that a bus has the same context as a car but not vice versa: the context of a car *includes* the context of a bus. This is supported by subsequent experiments.

Finally, in the person row nothing changes at all. This suggests that for the person class the context is not learned, *i.e.* a person is context free in this dataset.

4.1.3 Context Patches Only

In this experiment we use only `context` patches. The difference between the CAMP of the baseline experiment and this experiment is shown in figure 2b.

Using only context, the classification for all classes except *diningtable* goes down.

Furthermore, far fewer persons are confused as bottles, suggesting that the confusion mainly comes from the object patches. We will come back to this in the next experiment.

Dogs and cats are less confused, but this loss is approximately the same as the decline of Average Precision of both classes. This suggests that both share a similar context.

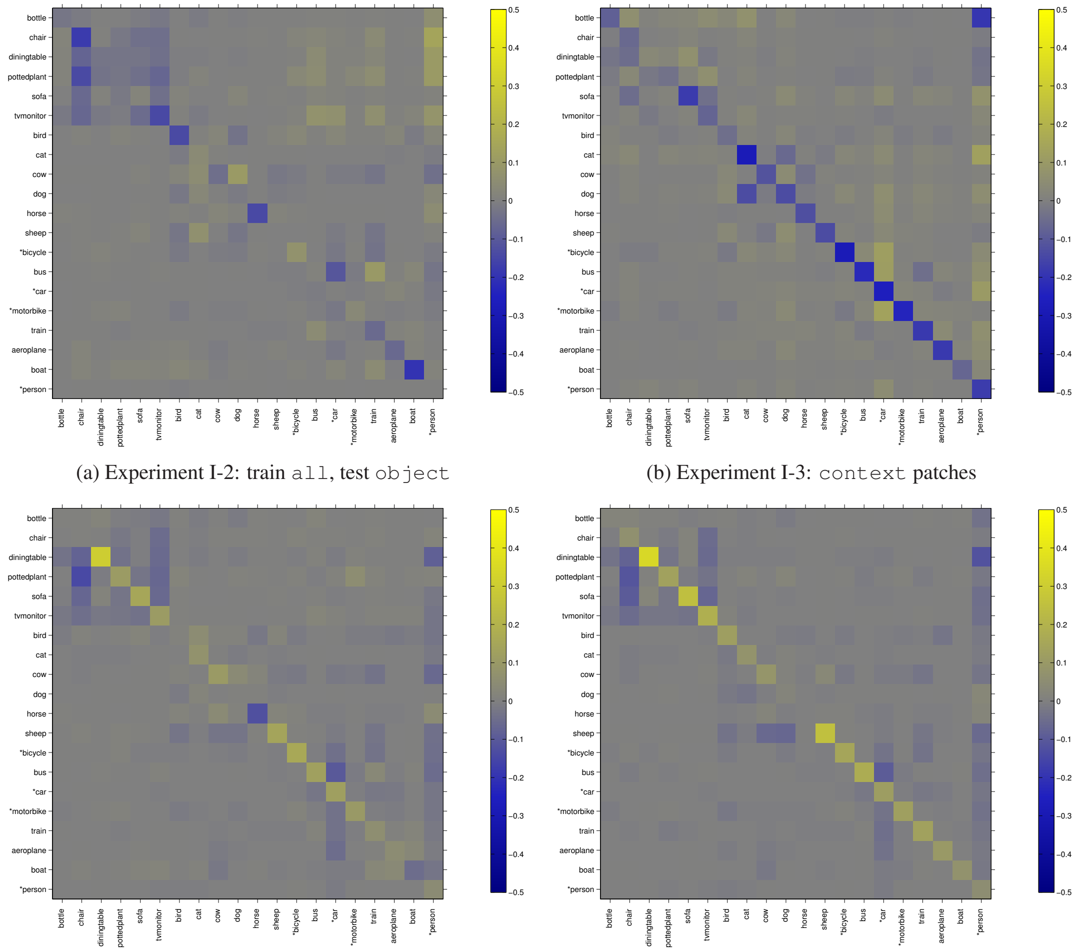
Cars are more often confused as other land-vehicles. The other way around this is not significant. This suggests that while the contexts of bicycle, bus, and motorbike are disjunct, the car context includes them all. For the bus class this confirms our earlier observations. Inspecting the top ranked results (data not shown), we see that the motorbike-context is dominated by a race circuit and the bus-context is dominated by urban environments. While cars are present in both of these contexts, buses rarely race and this dataset contains few motorbikes in urban environments. For bicycles we could not identify any apparent context.

4.1.4 Object Patches Only

In this experiment we learn and test on the `object` patches only. The difference between the baseline CAMP and the `object` patches only CAMP is given in figure 2c.

In general the performance increases when only the object patches are used: the Mean Average Precision over all classes is 0.54. This is 0.08 higher than using all patches. The *horse* and *boat* are the only classes that perform worse without context. For *boat* the water is obviously important evidence. For the *horse* class the jumping contest dominant in this dataset is important.

Persons are still as often confused with bottles as in the baseline experiment. This confirms the suspicions of the previous experiment that confusion here is caused by the `object` patches. This suggests that persons and bottles



(a) Experiment I-2: train `all`, test `object`

(b) Experiment I-3: `context` patches

(c) Experiment I-4: `object` patches

(d) Experiment II: `context+object` patches

Figure 2: Confusion Average Precision Matrices (CAMPs) for our various experiments. For clarity we plotted each CAMP after subtracting it with the baseline CAMP.

either co-occur often or look similar in SIFT terms. Indeed, in 59% of the *bottle* images there is a *person*. But a person occurs in 77% of the *horse* class, 66% of the *bicycle* class, and 69% of the *motorbike* class, and these classes do not exhibit as much confusion as the bottle. Therefore it is more likely that a person looks like a bottle in SIFT-space.

Cars are significantly less confused with other land-vehicles, suggesting that context is the source of the confusion in our baseline experiment.

For the person row nothing changes. Like in section 4.1.2, this suggests that a person is context free in this dataset and that this is learned by the classifier.

The experiments in this section show that context provides relevant information in this dataset. Therefore the result of [22] that context is superfluous does not generalise.

4.2. Context as an Extra Information Channel

Having established that context contains relevant information, we now test if we can use this information to increase performance. We do this by creating for each image a separate codebook frequency histogram for the `context` and `object` patches, which are normalized individually. Then we concatenate them to obtain the final codebook frequency histogram which is used as input into the SVM. The difference CAMP for this experiment is shown in figure 2d and looks similar to the `object`-patch only experiment of figure 2c. Therefore we will limit our discussion to table 1, which shows the actual Average Precision values of this experiment, the baseline experiment, and the individual information channels.

We observe an improvement of 0.12 in Mean Average Precision over the baseline experiment, which is a relative

	All	Context	Object	Object+Context
bottle	0.192	0.106	0.197	0.210
chair	0.449	0.390	0.454	0.514
diningtable	0.276	0.307	0.580	0.620
pottedplant	0.156	0.119	0.263	0.287
sofa	0.369	0.191	0.519	0.619
tv	0.409	0.337	0.516	0.595
bird	0.378	0.332	0.387	0.498
cat	0.437	0.155	0.508	0.511
cow	0.274	0.161	0.376	0.359
dog	0.367	0.234	0.374	0.386
horse	0.692	0.564	0.568	0.690
sheep	0.284	0.150	0.427	0.535
*bicycle	0.462	0.178	0.621	0.610
bus	0.496	0.272	0.624	0.668
*car	0.690	0.431	0.813	0.807
*motorbike	0.491	0.253	0.592	0.614
train	0.686	0.503	0.742	0.817
aeroplane	0.702	0.522	0.752	0.802
boat	0.623	0.555	0.574	0.696
*person	0.792	0.616	0.836	0.846
mean AP	0.461	0.319	0.536	0.584

Table 1: Average Precision scores for the baseline, `context` only, `object` only, and `object+context` experiment. The asterisks (*) denote classes used in [22].

improvement of 26%.

Furthermore, there is an improvement of 0.05 Mean Average Precision over the `object` patches only experiment. For the classes *sofa, bird, horse, sheep,* and *boat* this improvement is even higher than 0.10. There are no classes whose performance decreases significantly.

This experiment shows that there is an upper bound of improvement of 0.12 in terms of Mean Average Precision by treating `context` and `object` patches separately.

4.3. The Spatial Extent of an Object

We determine the best bounding box to recognise each object in the Pascal 2007 dataset, giving us its spatial extent. This is done by expanding and shrinking the bounding boxes denoting the location of the object, which are then used to make the `context`/`object` distinction as before. Specifically, we multiply the width and height of the bounding box with respectively 0.6, 0.8, 1, 1.2, and 1.4, while keeping the centre of the bounding box the same. The resulting area will cover respectively $36\%, 64\%, 100\%, 144\%,$ and 196% of the original bounding box.

We will determine the spatial extent in two settings: using only `object` patches, as is common in most object localization methods, and using `context` and `object` patches separately, which we have shown to give better results in the recognition task.

4.3.1 Spatial Extent using Object Only

Figure 3a shows the result for using only `object` patches for a representative set of objects and for the overall Mean Average Precision. We can distinguish three trends under which all but the *bird* class can be categorised:

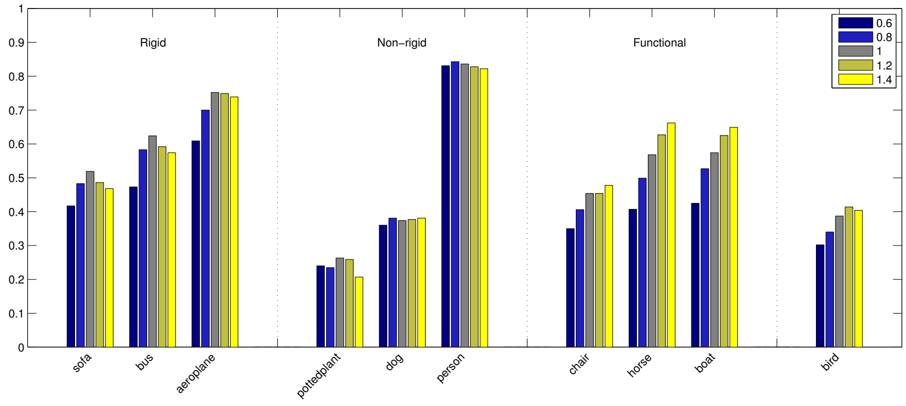
1. The object is best recognised if the bounding box fits tightly around the object. This is the case for *diningtable, sofa, tv/monitor, aeroplane, bicycle, bus, car, motorbike,* and *train*. The objects falling into this category are all *well defined, rigid* objects. All patches of these objects are informative.

2. For some objects the spatial extent has no clear border: all extents perform equally well. This is the case for *bottle, pottedplant, cat, cow, dog, sheep,* and *person*. The objects here are all *non-rigid* objects without a well-defined shape. Animals have no well-defined shape because of the high variability of poses they can assume. Plants grow in all shapes. This suggests a small part of the object is sufficient for recognition. Adding extra context does not significantly hurt performance.

3. More spatial context is better. This is the case for *chair*, *horse*, and *boat*. These object classes all have a high variability in appearance and are mainly classified by their *function*. A boat can be characterized as a vehicle travelling through water. The function of a chair is to sit. In this dataset a horse is used for jumping competitions. Their function highly restricts their context: for a boat this is water, for a chair this is a specific indoor environment such as a living or conference room, and for the horse this is the jumping contest.

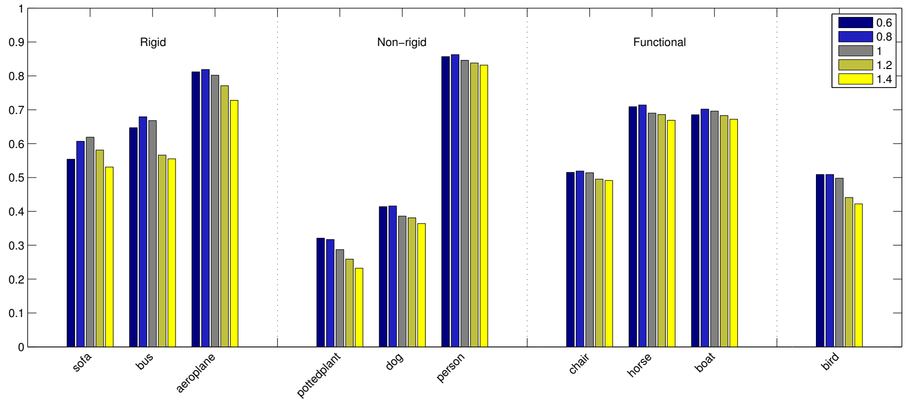
4.3.2 Spatial Extent using Context and Object

Results for the spatial extent when using both `context` and `object` patches are shown in figure 3b. With the inclusion of `context` information the rigid objects retain the same behaviour as before. For non-rigid objects small bounding boxes perform best and enlarging them gradually decreases performance. This supports the theory that a small part of the object is sufficient for recognition. The preference for a small box is probably due to the exclusion of stray context patches. For functional objects there is little change; the context is highly important, but is either captured by the `context`-patches for smaller bounding boxes or captured by `object`-patches for larger bounding boxes.



Multiplication Factor	Mean AP
0.6	0.457
0.8	0.505
1	0.536
1.2	0.527
1.4	0.519

(a) `object` patches only



Multiplication Factor	Mean AP
0.6	0.583
0.8	0.594
1	0.584
1.2	0.553
1.4	0.528

(b) `context+object` patches

Figure 3: The Spatial Extent of an Object for a selection of classes. The various bars per class represent the multiplication factor of the length and width of the bounding box. The tables provide the Mean Average Precision scores.

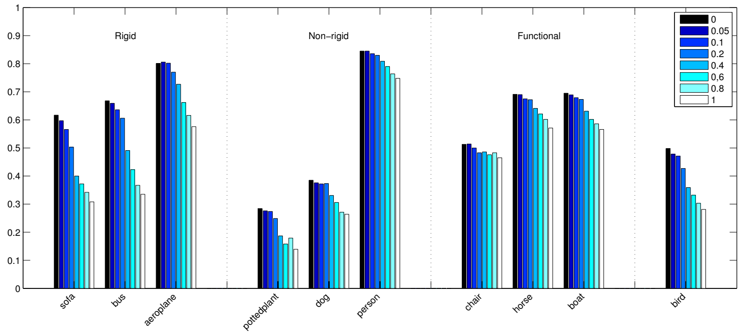
In general, using both the `context` and `object` patches results in less sensitivity to the box size. This especially holds for small boxes.

4.4. The Influence of Localisation Accuracy

In a final experiment, we test how accurate localisation should be in order to gain improvements in performance on the recognition task. This is done by adding an error to the location of the bounding box in the test phase only. We draw the error from a Gaussian distribution with respect to the length and width of the bounding box, using standard deviations of 0, 0.01, 0.05, 0.1, 0.2, 0.4, 0.6, 0.8, and 1; a standard deviation of 1 means that on average the estimated location of the object is exactly besides its true location while 66% of these boxes still have some overlap with the true bounding box. Like before we use `object` and `context` patches separately. Figure 4 presents our results.

We see that recognition accuracy increases heavily with a more precise localisation of the `object` patches. A localisation error of 0.4 results in an accuracy approximately equal to the baseline experiment, where we observe a significant benefit only for *diningtable*, which increases 0.14 over the baseline at this error rate. For a localisation error of 0.2 we get a Mean Average Precision of 0.53, which is already 0.07 higher than the baseline.

The results suggest that only precise localisation of the `object` patches gives a performance increase. Therefore a more precise localisation, for example by using a segmentation, is likely to yield an even better performance than the bound presented in this paper.



sigma	MAP
0	0.583
0.05	0.574
0.1	0.564
0.2	0.530
0.4	0.474
0.6	0.440
0.8	0.416
1	0.387

Figure 4: The influence of Gaussian localisation errors on Mean Average Precision

5. Conclusion

We have shown that context is important in a Bag of Words framework for this dataset, hence the conclusion of Zhang *et al*. [22] that context is superfluous in this framework does not generalise. But as this paper also used a single dataset, further experiments are needed to verify if our results do generalise.

Concerning the spatial extent of an object we can distinguish three classes in the Pascal 2007 dataset: (I) Visually well defined, rigid objects such as *aeroplane* and *tv/monitor*. The spatial extent of these classes include the complete object. (II) Non-rigid classes like *pottedplant* and *dog*. Their spatial extent is unbounded: a small part of the object already suffices for recognition, but some additional context does not hurt. (III) Functional objects such as *chair* and *boat*. These classes have a large variability in appearance and rely heavily on their context for recognition. Their preferred spatial extent is the whole image.

Finally, localising the `object` patches with a bounding box, if done precisely, gives an upper bound of 26% relative improvement in terms of Mean Average Precision for our dataset. Using a segmentation rather than a bounding box for localisation is likely to give higher accuracy.

References

[1] M. Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5:617–629, 2004.

[2] I. Biederman. *Perceptual Organization*, chapter On the semantics of a glance at a scene, pages 213–263. Lawrence Erlbaum, Hillsdale, New Jersey, 1981.

[3] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *CVPR*, 2007.

[4] M. Everingham and J. Winn. The Pascal VOC 2007 development kit. Technical report, University of Leeds, 2007.

[5] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV*, 2008.

[6] Y. Jiang, C. Ngo, and J. Yang. Towards optimal bag-of-features for object categorization and semantic video retrieval. In *CIVR*, 2007.

[7] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *ICCV*, 2005.

[8] C. Lampert, M. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR*, 2008.

[9] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[10] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.

[11] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning representations for visual object class recognition. Pascal VOC 2007 challenge workshop. ICCV, 2007.

[12] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 27:1615–1630, 2005.

[13] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In *NIPS*, 2006.

[14] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[15] E. Nowak, F. Jurie, and B. Triggs. Sampling Strategies for Bag-of-Features Image Classification. In *ECCV*, 2006.

[16] D. Parikh, C. Zitnick, and T. Chen. From appearance to context-based recognition: Dense labeling in small images. In *CVPR*, 2008.

[17] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *ICCV*, 2007.

[18] A. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVID. In *MIR*, 2006.

[19] A. Torralba and A. Oliva. Statistics of natural image categories. *Network: Computation in Neural Systems*, 14:391–412, 2003.

[20] T. Tuytelaars and C. Schmid. Vector quantizing feature space with a regular lattice. In *ICCV*, 2007.

[21] K. van de Sande, T. Gevers, and C. Snoek. A comparison of color features for visual concept classification. In *CIVR*, 2008.

[22] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73:213–238, 2007.