Towards autonomous adaptation in visual tasks

Boyan Bonev and Miguel Cazorla

Dept. de Ciencias de la Computación e Inteligencia Artificial
Universidad de Alicante
Apdo. 99, E-03080 Alicante
{boyan,miguel}@dccia.ua.es

Abstract

In this paper we study an appearance-based visual recognition approach. The only features that we extract from the images are basic filters like some edge and corner detectors, and color filters. This idea is biologically inspired. We construct a bank of filters and automatically select a small feature set which is appropriate for the given recognition task. We present interesting results about Feature Selection showing that very small feature sets can yield better classification performance than the complete set. We analyze this method's generality and performance in indoor and outdoor environments.

1 Introduction

A fundamental problem in mobile robotics is localization. Many approaches solve this problem in constrained environments using different kinds of sensors, artificial and even natural landmarks. The rapid development of computer vision is causing great interest in autonomous robotics. Cameras provide much richer information than range sensors like laser beams and sonars, informing about textures and illumination. However, the correct interpretation of that information is a complex problem which still remains unsolved.

Structural-description models and image-based models are two well-differentiated approaches to visual recognition. In a structural description objects and environments are represented as 3D or 2D parts which could have position in the space and volume, as well as relations with other parts of the scene. This kind of data is very useful for mobile robotics. Though, to retrieve such high-level information from an image has proved to be a hard task, where computationally expensive algorithms are involved. Moreover such systems are very sensitive to illumination variations, noise, and other environmental changes.

Figure 1: Examples of a) the indoor view and b) the outdoor view, obtained with the omnidirectional camera.

Image-based recognition is causing great interest in computer vision. In robotics it has received most attention in visual navigation. Jogan and Leonardis [6], for example, construct an omnidirectional appearance-based model of the environment for localization, with reduced dimensionality (PCA) and a mechanism for rotational invariance. Menegatti et al. [4] weight omnidirectional samples according to similarity among images, to implement then a Monte-Carlo localization for indoor environments. On the other hand, appearance-based approaches tend to be more suitable for outdoor and arbitrary environments (like those in
Figure 1), as we will see along this paper.

In this work we study an appearance-based visual learning method by extracting low-level information from omnidirectional images. Such information is extracted by applying basic filters to the image (edge detectors, color filters, etc). Biological evidences [12], [16] report that low-level filters play an important role in biological visual recognition systems. For example, Gabor filters model the visual processing carried out by the simple and complex cells of the primary visual cortex of higher mammals. The organization of these cells results from an unsupervised learning in the visual system, during the first months of life [5]. In computer vision for object recognition Gabor filters [11], Haar features, steerable filters [10] and color co-occurrence histograms [2], [13] are being widely used, providing tolerance to noise and robustness to small perspective changes and occlusions. These methods tend to be probabilistic and machine-learning based, and they are not restricted to any particular environment, nor to some definite set of filters [15].

This generality is a motivation for the present study. Our purpose is to train a general image-based learning system (explained in Section 2) to recognize a part of the environment from one omnidirectional image (Figure 1). We collect a large bank of low-level filters (Section 3) and automatically select the most appropriate set for any particular environment. This selection is performed with supervised learning and Wrapper Feature Selection [8] (as explained and illustrated in Section 4). The computational complexity of this method does not depend on the image size, but on the number of filters used to evaluate an image. The filters bank could have many more filters than the selected set, however, the complexity of the Feature Selection depends on the size of the filters bank. In this paper we demonstrate that a very small number of filters is sufficient for many classification tasks, moreover, a large number of features usually worsens the results.

We have a lot of future work (Section 5) related to this method, such as online feature selection (adding filters during the learning process) and online machine learning (adding samples during the learning process).

Finally, a possible application is "the kidnapped robot problem" in which the robot is placed in a random position of the map and is turned on, in other words, to supply robot's localization with a higher level information ("You are in the living room, near to the window.") than its sonars could provide (Figure 9). Another possible application is to train the system for recognizing poorly defined natural landmarks.
2 Experimental setup

In this paper we analyze the data of two different experiments: indoor and outdoor (as shown in Figure 1). We use a photo camera with omnidirectional lens in order to receive much more information from the environment than a local view would provide us. In this way the only problems that remain are occlusions and noise. The resolution is 400 × 400 but it could be any other, as the only information treated is the filters' response, and it does not depend on the size of the images.

Images were taken a) along the corridors of a building and b) around a building. The corridors and the route around the building are represented in the Figure 2. The number of images taken were 70 for each one of the experiments. The physical distance between pictures is roughly 1.50m for the indoor experiment and 3m for the outdoor experiment.

When the (labelled) images are collected, we process them using the filters bank, in order to learn information from them. This is the training phase, where Feature Selection takes place. We explain this process in the following sections.

The constructed system consists of a set of selected filters and a trained classifier. The system is capable of identifying the label of a new (unlabelled) image. The only computational cost relies upon the number of selected filters.

3 Feature extraction

The filters we use have a low computational cost, and are rotation invariant. The color filters are calculated in the HSB color space in order to maintain some tolerance to lighting variations. The 17 filters are:

- Nitzberg
- Canny
- Horizontal Gradient
- Vertical Gradient
- Gradient Magnitude
- 12 Color Filters $H_i$, $1 \leq i \leq 12$

Some of them are redundant, for example the magnitude and the gradients. Others are similar, like Canny and gradient's magnitude. Finally, some filters may overlap, for example the color filters. The color filters return the probability distribution of some definite colour $H$ (from the HSB color space).

These filters are applied to four different parts of the image, as shown on the Figure 3. The image is divided in four concentric rings to keep rotation invariance (the idea comes from the Transformation Ring Projection [17]). This division also provides additional information by establishing different features for different height levels of the environment. For example, the inner ring has information only about the nearest floor, while the outer ring informs about buildings, trees and sky, but not about the floor.

Finally, when we apply a filter to a ring, we obtain a histogram. We performed experiments with 2, 4, 12 and 256 bins in the histograms. As the first bin contains the number of pixels which have no response to the

![Figure 3: An image divided in concentric rings notated as ring 1, ring 2, ring 3 and ring 4, respectively.](image)

![Figure 4: The feature extraction process.](image)
filter, we do not add it to the feature set. So each feature contains the value of a definite histogram’s bin, of a definite filter, applied to one of the four rings of the image. The whole process is illustrated on the Figure 4. The total number of features $N_F$ per image is

$$N_F = C \times K \times (B - 1) \quad (1)$$

where $C$ is the number of rings (4), $K$ the number of filters (17) and $B$ is the number of bins.

4 Machine Learning

In this section we present classification results with supervised learning (Subsection 4.1), compare them to some clustering analysis (Subsection 4.2) and optimize the filters set (Subsection 4.3). The experiments are performed with a k-Nearest Neighbor classifier because lazy learners are adequate when the amount of training data is not large. We selected $k=1$, however $k=3$ and $k=5$ also performed well. For all experiments the classification error reported is calculated using 10-fold Cross Validation. This process consists of randomly dividing the data in 10 equal parts, and constructing the classifier 10 times, each time leaving one of the 10 parts for testing, and the rest for training. The average classification error is finally reported.

4.1 Supervised learning

We performed a manual labelling of the different parts of each environment, which is represented on the Figure 2. The classes we establish are not clusters inherent to the data, but are human-established. For example one class could be, "The beginning of the corridor with blue columns", another could be "Near to the entrance".

On the Table 4.1 we can see classification errors for experiments with different number of classes. For higher number of classes we can expect worse classification error. We can also see that 2-bins histograms present some lack of information for several experiments. On the other hand, 256-bins histograms are excessive and do not improve results. We estimate that using 4 bins is reasonable. With 12 bins the amount of features is very large and Feature Selection is more costly.

| Table 1: Errors in % : Supervised Learning - Indoor and Outdoor experiments. 10-fold Cross Validation errors calculated for experiments labelled with 2, 4, 8 and 16 classes (L), and histograms with 2, 4, 12 and 256 bins. |
|---|---|---|---|---|
| | 2 | 4 | 12 | 256 |
| indoor | | | | |
| 2 | 8.809 | 9.142 | 9.142 | 5.619 |
| 4 | 11.571 | 9.190 | 8.714 | 15.714 |
| 8 | 30.571 | 22.381 | 26.952 | 26.904 |
| 16 | 49.000 | 51.667 | 60.571 | 55.905 |
| outdoor | | | | |
| 2 | 1.392 | 2.785 | 1.964 | 5.500 |
| 4 | 4.714 | 5.785 | 3.892 | 7.142 |
| 8 | 12.428 | 7.964 | 12.500 | 18.464 |

4.2 Clustering analysis

We also analyzed clusters in the data with the K-means algorithm for different number of classes. We observed that some of the clusters coincide with the classes we establish (with some offset) and other do not agree. Disagreements are usually due to some object present in the image, shades and light. We calculated that for the 8-classes experiment, the percentage of disagreement is 14% indoor, and 19% outdoor. For the 16-classes indoor experiment the labels and the clustering disagree in 27%.

In the following section we will show how classification error percentages change after Feature Selection.

4.3 Feature Selection

Feature Selection is a field with increasing interest in Machine Learning. The literature differentiates among three kinds of Feature Selection: Filter method [8], Wrapper method [14], and On-line [9]. Filter Feature Selection does not take into account the properties of the classifier, as it performs statistical tests to the variables, while Wrapper Feature Selection
tests different feature sets by constructing the classifier. On-line Feature Selection adds features during the selection process, and forms part of the future work of the present study. In the Figure 5 we have outlined the Wrapper Feature Selection process of our system.

There are different strategies for generating feature combinations. The only way to assure that a feature set is optimum is the exhaustive search among feature combinations. Of course, the 
\textit{curse of dimensionality} limits this search, as its complexity is

\begin{equation}
\mathcal{O}(n) = \sum_{i=1}^{n} \binom{n}{i} \quad (2)
\end{equation}

If we model the filters response with histograms of only 2 bins, we have a total of 68 features, and the total number of combinations is \(2.9515 \times 10^{20}\). It would take trillions of centuries to evaluate all of them. Anyway we have searched exhaustively among the subset where \(N_F = 3\) (i.e. feature sets with cardinality 3).

The best three features for the 8-classes indoor experiment are: Canny, Color Filter 1, and Color Filter 5, all of them applied to the ring 2 (notation illustrated in the Figure 3). The Cross Validation (CV) error yielded with only 3 features is 24.52\%, much better than the 30.57\% CV error yielded by the complete set of 68 features.

For the outdoor experiment the best three features are Color Filter 9 on ring 2, Nitzberg and Color Filter 8 on ring 3. In this case 3 features are not enough (see Table 2) for decreasing the error, as this experiment needs more features to be selected (graphically illustrated on Figure 6).

The fastest way to select from a large amount of features is a greedy strategy. Its computational complexity is

\begin{equation}
\mathcal{O}(n) = \sum_{i=1}^{n} i \quad (3)
\end{equation}

and the algorithm is as follows:

\begin{verbatim}
DATA_{M \times N_F} ← vectors of all \((M)\) samples
FS = ∅
F = \{feature_1, feature_2, ..., features_{N_F}\}
while F ≠ ∅
    ∀i \(\mid\) feature_i ∈ F
    DS = DATA(FS ∪ \{feature_i\}) /* therefore DS ∈ DATA */
    E_i = 10FoldCrossValid(DS)
    selected = arg min_i E_i /* and also store E_i */
    FS = FS ∪ \{feature selected\}
    F = F \sim \{feature selected\}
end
\end{verbatim}

At the end of each iteration a new feature is selected and its CV error is stored. By plotting these CV errors we obtain graphics of the classification errors for different number of features, from 0 to \(N_F\). See Figure 6 with 8-classes indoor and outdoor experiments, using 4 bins histograms (i.e. a total of 204 features, as already explained in Section 3). We can see that the indoor problem needs just 45 features (out of 204) to obtain a good classification (detailed results on Table 2). The outdoor problem needs 138 features for a better performance. This is due to a higher complexity of the natural environment, and the presence of noise coming from people and several objects.

A larger number of bins can (but not necessarily) improve the classification. In this case the use of 12 bins (744 total features) do improve it, while 2 bins (68 total features) do worsen classification. See figure 7 where this difference is illustrated. On the Table 2 are represented several feature selection results using different number of histogram bins.

Another important factor is the number of classes. As the number of classes increases, the
classification performance decays, and more features are needed. In the Figure 8 we compare feature selection performance on the 8-classes and the 16-classes experiments.

Finally it is worth commenting out some statistics about the selected filters. The rings whose features are selected are usually the ring 2 and ring 3. The ring 1 and the ring 4 always have a little lower number of selected features. On the other hand when a bin from a definite filter is selected, it is highly probable that the rest of the bins of that filter will also be selected. Another interesting phenomenon is that in some cases the Nitzberg, Canny, Horizontal Gradient, Vertical Gradient, and Gradient Magnitude are selected all together, even though some of them are redundant. However, if we force the number of selected features to

Figure 6: Comparison of the Feature Selection results for the indoor and outdoor 8-classes experiments.

Figure 7: Comparison of Feature Selection using 2 bins and 12 bins histograms, on the 8-classes indoor experiment.

Figure 8: Comparison of 8-classes and 16-classes in the same environment (indoor). The 16-classes experiment reports a higher errors but similar performance evolution.
Table 2: Cross Validation errors without and with Feature Selection, as well as the number of selected features for several experiments indoor and outdoor, with different number of classes (L) and different number of bins (2 bins for 68 feat., 4 bins for 204 feat. and 12 bins for 748 feat.)

<table>
<thead>
<tr>
<th>L</th>
<th>All Features</th>
<th>Feature Sel.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Feat. Err.%</td>
<td>Feat. Err.%</td>
</tr>
<tr>
<td>8</td>
<td>68 30.57</td>
<td>3 21.52</td>
</tr>
<tr>
<td>8</td>
<td>68 30.57</td>
<td>38 13.33</td>
</tr>
<tr>
<td>8</td>
<td>204 22.38</td>
<td>45 7.51</td>
</tr>
<tr>
<td>8</td>
<td>748 26.95</td>
<td>234 3.33</td>
</tr>
<tr>
<td>16</td>
<td>68 49.00</td>
<td>31 26.43</td>
</tr>
<tr>
<td>16</td>
<td>204 51.67</td>
<td>66 22.86</td>
</tr>
<tr>
<td>8</td>
<td>68 12.42</td>
<td>3 16.07</td>
</tr>
<tr>
<td>8</td>
<td>68 12.42</td>
<td>19 3.93</td>
</tr>
<tr>
<td>8</td>
<td>204 7.96</td>
<td>138 1.42</td>
</tr>
<tr>
<td>8</td>
<td>748 12.50</td>
<td>184 1.25</td>
</tr>
</tbody>
</table>

be low, this does not occur. For three features always the selected features are some edge detector and two color filters.

About the differences between the indoor and the outdoor experiments, it can be said that the indoor environment requires less different color filters (roughly 2 colours). For the ring 2, only color filters are selected, while for the ring 3 the selected are edge detectors. In the outdoor case, the selected features are more homogeneously distributed among rings and filters.

5 Conclusions and further work

In this work we analyze an appearance-based image recognition method, using filters of low computational cost. The main idea we introduce is to apply Feature Selection technique to a large bank of filters. We show how a very small feature set is capable of classifying with better results than the complete feature set. This results in a real-time computational cost for recognition, although the offline feature selection process has a very high complexity.

We tested the method in two different environments, explaining the differences between them. We work on a general system which can be trained for a variety of vision tasks, and not only indoor or artificial setups.

Such a system has a lot of future work. In the first place we want to improve feature selection by using online techniques. We want to enable the system to keep on selecting and adding (or removing) features, while classifying images. Many filters can be calculated in parallel, so the real complexity problem is centered in the Feature Selection.

The omnidirectional camera (Figure 9) gives us a lot of visual information and is widely used [3] in mobile robotics, however different visual sensors could be tested. Another test to perform is to train the system with images from the same environment in different conditions (morning, afternoon, rain, electric light) and analyze which of the filters will be selected and what the recognition rate will be.

Finally, in this work we have not considered the perceptual aliasing problem. It requires higher level techniques such as considering sequences. Due to the low level of abstraction at which the described method works, it has clear limitations and its performance in combination with localization techniques has to be studied.

Figure 9: A possible application: camera with omnidirectional lens installed on a Magellan Pro platform.
References


