# Learning Multi-class Topological Mapping using Visual Information

Anna Romero and Miguel Cazorla

*Dpto. of Artificial Intelligence, University of Alicante., P.O. Box 99. 03080, Alicante, Spain*

Keywords: Topological Mapping, Graph Matching, Visual Features, Image Segmentation.

Abstract: Mapping of an unknown environment is an important area within robotics. The map obtained can be used in more complex problems such as localisation, scene recognition, navigation, SLAM, etc.. Topological maps, inspired by the human mental description of their surroundings, do not seek for accurate measures, but for the classification of the real environment in areas containing distinctive features that differentiate them from other areas. The use of learning techniques can help us to define different areas of the environment so that the robot can recognise them later. In this paper, we propose the use of Samme algorithm, a supervised learning method based on AdaBoost to select the best visual features that describe each area of a topological map.

## 1 INTRODUCTION

Scene recognition and topological localisation is an emergent area of robotics as it helps in more complex activities such as autonomous navigation, performing certain tasks (eg., transport of objects, target tracking), SLAM (Simultaneous Localisation and Mapping), and so on. In most of the literature performing scene recognition or localisation needs some image processing so that we extract distinguishable features and if they reappear we are confident to find them in another image. One of the most difficult tasks within the visual mapping is to identify the features (among all detected, that can be thousands) to better define the environment.

Depending on the kind of algorithm used to solve the problem of scene recognition, we have supervised or unsupervised methods. When the algorithm does not require pre-existing database, the map is constructed as the robot navigates through the environment as in (Liu et al., 2009). In our recent work (Romero and Cazorla, 2012b), we propose a similar approach to solve the problem using an unsupervised method for building a topological visual map.

Supervised algorithms are those where input data has been previously classified (by hand or other method). In the case of visual recognition of scenes, the algorithm needs a database where the images are classified by category (or area of the environment) to which they belong. In (Wu et al., 2009), they introduce the concept of “Visual Place Categorization” (VPC) which consists of identifying the semantic category of one place/room using visual information. In this paper we propose the use of the AdaBoost based SAMME (Stagewise Additive Modelling using a Multi-class Exponential loss function) algorithm (Zhu et al., 2005) to learn topological maps manually calculated and using them in scene recognition problems.

AdaBoost (Adaptive Boosting) is a machine learning algorithm presented in (Freund and Schapire, 1997). The algorithm constructs a strong classifier from weak classifiers (landmarks or features in visual problems) that are not able to make a reliable classification. The solution has been widely successfully used in the two-class classification problems, but in terms of multi-class classification, most of the boosting algorithms have reduced the problem to treat multiple two-class problems as in (Freund and Schapire, 1997) and (Friedman et al., 2000). The SAMME algorithm, proposed in (Zhu et al., 2005) is a modification of the original AdaBoost algorithm to treat the multi-class problem without simplifying it to two-class classification problem. AdaBoost has also been successfully applied in topological mapping problems as in (Mozos and Burgard, 2006) using laser range data.

In this paper, we propose the combination of the SAMME algorithm and MSER (Maximally Stable Extremal Region) detector (Pajdla et al., ) and SIFT (Scale-Invariant Feature Transform) descriptor (Lowe, 2004) to develop a scene recognition algorithm. In the literature we can find several algorithms that propose to use AdaBoost with invariant features,

Romero A. and Cazorla M..
Learning Multi-class Topological Mapping using Visual Information.
DOI: 10.5220/0004297103160321
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 316-321
ISBN: 978-989-8565-48-8

as in (Yin and Xie, 2007), which proposes a method to learn hand postures and (Hu et al., 2008) which uses color information and SIFT features to re-detect people. SAMME method is adapted in order to manage SIFT descriptor. Furthermore, we compare the SAMME results with a topological mapping adaptation of the Viola-Jones algorithm where the problem is reduced to multiple two-class problems.

The paper is structured as follows: In Section 2 we describe the SAMME algorithm based on AdaBoost and the modifications proposed to apply the method to the visual scene recognition problem. In Section 3, we will present some results and the comparison between the results of the Viola-Jones based algorithm and the proposed SAMME based algorithm. Finally, in Section 4 we will draw some conclusions and the possible future work.

## 2 AdaBoost

The AdaBoost (Adaptive Boosting) algorithm ((Freund and Schapire, 1997)) is a machine learning meta-algorithm as it can be used in conjunction with other learning algorithms to improve their performance. In order to do this, AdaBoost combines a collection of weak classification functions (for example simple perceptrons) to form a stronger classifier, following the philosophy of the boosting algorithms. The algorithm is adaptive as it selects a new weak classifier "specialised in those examples that have been misclassified by all the previous weak classifiers. At each round of the algorithm, it is selected a weak classifier that have less error (*i.e.* selecting the classifier that best distinguishes between positive and negative examples). The examples have an associated weight that emphasize those which have been wrongly catalogued, so the next classifier will have a minor error if it assigns the correct class to those examples. The final strong classifier is a weighted combination of weak classifiers followed by the used threshold.

### 2.1 SAMME Algorithm

AdaBoost has been proved to be a good solution to the two-class classification, however it is not the case for the multi-class problem. In the two-class problem, it is required that the error of each weak classifier to be a slightly better than a two-classes random guess *i.e.*, $\varepsilon_t \leq 1/2$. Otherwise $\alpha_t$ will be negative and the weights of the training samples will be wrongly updated. However when $K > 2$, accuracy $1/2$ may be much harder to achieve than the random guessing accuracy rate $1/K$. Therefore, the AdaBoost original algorithm may fail when the weak classifier $h_j$ has not been correctly selected.

The SAMME algorithm (proposed in (Zhu et al., 2005)) is a multi-class boosting method that allows the classification of more than two different classes by relaxing the error condition. Algorithm 1 shows the pseudo code for the SAMME algorithm.

**Algorithm 1:** SAMME: Stagewise Additive Modeling using a Multi-class Exponential loss function.

```
procedure LEARNINGFEATURES(Image examples: $(x_1,y_1),...,(x_n,y_n)$)   ▷ where $y_i = 0,1$ for negative and positive examples respectively
    $w_i = \frac{1}{n}$   ▷ $i = 1,2,...,n$
    for $t \leftarrow 1,...,T$ do
        For each feature $j$, train a classifier $h_j$ with only one feature. The error made is calculated in function of $w_t$: $\varepsilon_j = \frac{\sum_{i=1}^{n} w_i \cdot \mathbb{I}(y_i \neq h_j)}{\sum_{i=1}^{n} w_i}$   ▷ $\mathbb{I} = 1$, if $(y_i \neq h_j)$; $\mathbb{I} = 0$, otherwise
        Choose the classifier $h_t$ with the lowest error $\varepsilon_t$
        $\alpha_t = \log\frac{1-\varepsilon_t}{\varepsilon_t} + \log(K-1)$
        $w_i \leftarrow w_i \cdot exp(\alpha_t \cdot \mathbb{I}(y_i \neq h_j))$   ▷ $i = 1,2,...,n$
        Re-normalize $w_i$
    end for
end procedure

procedure STRONGCLASSIFIER(Image to classify: $x$)
    $h(x) = \arg\max_{k} \sum_{t=1}^{T}(\alpha_t \cdot \mathbb{I}(h_t(x) = k))$   ▷ $\mathbb{I} = 1$, if $(h_t(x) = k)$; $\mathbb{I} = 0$, otherwise
end procedure
```

The SAMME algorithm is very similar to AdaBoost but takes into account the existence of $K$ classes in the problem, so that the error will now be greater than a random guessing among the $K$ classes, *i.e.* in order for $\alpha_t$ to be positive, the algorithm only needs $(1 - \varepsilon_t > 1/K)$. This change will give greater weight to the misclassified examples than in AdaBoost, and the combination of the weak classifiers will be a little different to the original method. Note that when $K = 2$ the SAMME algorithm becomes the original AdaBoost.

### 2.2 Adaptation of the SAMME Algorithm to Use Invariant Point Features

The algorithm SAMME allows to use any type of weak classifier provided that the $h_j$ formula is established. The equation indicates whether or not a feature (x) appears in a particular example. In our case, we intend to train a set of strong classifiers to describe each node in the topological map manually tagged described in Figure 1. To do this, we propose using the MSER detector with the SIFT descriptor as weak classifiers.

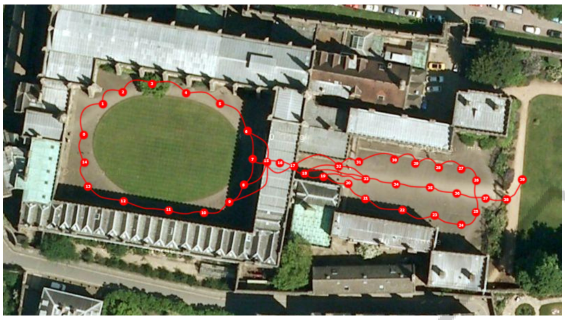

Figure 1: Topological map manually tagged. The nodes describe different areas of the environment and the edges the neighborhood relationships between nodes.

SIFT descriptors provide a vector of 128 size which describes the neighborhood of a point in the image invariant to scale, rotation and partially illumination changes. This descriptor can be compared to other descriptors using the Euclidean distance $D(P,Q) = \sqrt{\sum_{i=0}^{n}(p_i - q_i)^2}$, where $P$ and $Q$ are the two descriptors to compare. Using this distance we can see if two features appearing in two different images are actually the same feature, so that we can construct the weak classifier based on Euclidean distance. For that, we need to set the maximum threshold to consider that two features are equal. To automatize the process and not rely on hand-selected values, we propose the calculation of the threshold as follows:

$$Threshold(P) = M_d(P) + 3\sigma(P), \qquad (1)$$

where $M_d$ is the sample median of the distances between $P$ feature and the matched features for all positive examples, and $\sigma$ is the standard deviation of this distribution.

Therefore, the equation $h_j(x)$ would be as follows:

$$h_j(x) = \begin{cases} 1, & \text{if } D(f_j, f_x) \ < Threshold(f_j) \\ 0, & \text{otherwise} \end{cases} \qquad (2)$$

Furthermore, in Algorithm 1 the method for constructing a strong classifier selects the class with the higher sum of $\alpha_t$. However the algorithm not takes into account the sum of all alphas ($\alpha_t$), so that in the following situation:

$$Class\ 1: \begin{cases} \sum_{t=1}^{T}(\alpha_t \cdot \mathbb{I}(h_t(x) = k)) = 0.6 \\ \sum_{t=1}^{T}(\alpha_t) = 2.5 \\ Confidence\% = 24\% \end{cases} \qquad (3)$$

$$Class\ 2: \begin{cases} \sum_{t=1}^{T}(\alpha_t \cdot \mathbb{I}(h_t(x) = k)) = 0.4 \\ \sum_{t=1}^{T}(\alpha_t) = 1.4 \\ Confidence\% = 28.57\% \end{cases} \qquad (4)$$

the SAMME algorithm would select “Class 1” (data in 3) because the sum of $\alpha_t$ is higher than the second class (data in 4). However, the sum of all alphas is higher in “Class 1”, so that the confidence of the algorithm that the image belongs to that class is less than the confidence in “Class 2”. In our case, we work with the percentages to take into account this problem. With this information we will check which strong classifier is more confident that the input image belongs to its class, *i.e.* the image belongs to the node with the greater result that it gets in absolute terms (using the percentage of the total weighted value for the comparison). In Algorithm 2 we could see the pseudo-code describing the new definition of strong classifier and the scene recognition method proposed in this paper.

**Algorithm 2:** Scene Recognition based on SAMME algorithm.

```
procedure StrongClassifier(Image to classify $x$)
    $h(x) = [\sum_{t=1}^{T} \alpha_t h_t(x), \sum_{t=1}^{T} \alpha_t]$
end procedure

procedure SceneRecognition(Images to classify $x_1,...,x_N$)
    for $i \leftarrow x_1,...,x_N$ do
        $maxPercentage \leftarrow 0$
        for $node \leftarrow node_1,...,node_M$ do
            $[\alpha, \alpha_{total}] \leftarrow h_{node}(i)$
            $percentage \leftarrow \frac{\alpha}{\alpha_{total}} 100$
            if $percentage > maxPercentage$ then
                $maxPercentage \leftarrow percentage$
                $currentNode \leftarrow node$
                $Node_i \leftarrow node$
            end if
        end for
    end for
end procedure
```

As the algorithm has been set out, an input image can be assigned to a node of the map or even none, which would mean that we are in an unknown area (not part of the map, and therefore it has not been learned).

## 3 RESULTS

This section shows the results of applying the whole algorithm on the set of images described in paper

(Smith et al., 2009) which are available for download from the authors' website. The images are omnidirectional, with a resolution of 1024x309 (we have scaled the images in 50%). The tests were conducted on the images for the first route, the first 3000 of the dataset. In Figure 2 shows the path taken by the robot for the 3000 first images from the database. As we can see, it makes several loop closures and also has several hot spots, like the tree (lots of texture and occlusions) and the tunnel (very dark images with little information).

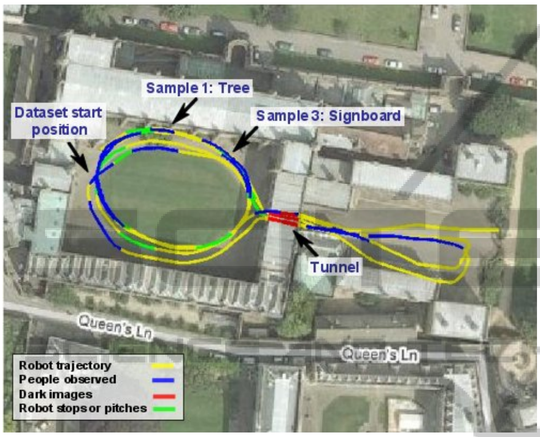

Figure 2: Robot path with several laps and hot spots (tunnel and tree areas). This image is from the web of the New College DataSet authors.

As mentioned above, the features used in the algorithm have been extracted with the MSER detector, which describes extreme regions whose intensity is maximum or minimum with respect to its neighboring regions. In Figure 3 we can see an image of the database with MSER features identified by an ellipse that is as close as possible to the detected region.

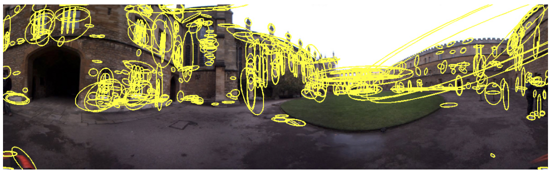

Figure 3: Image 1 of the NewCollegeDataSet with its MSER features.

The algorithm presented requires positive (belonging to the class, or node in our case) and negative examples. As positive examples we use the images of the node belonging to the first round (the average number of positive images are 10-15 per node). For negative examples, we take the 20 immediately preceding images to the node and the 20 immediately after, so we are looking for features that allow us to distinguish between the node and its neighboring nodes, *i.e.* features that are within the node, but not in the nearby nodes. The features obtained have an error not too small, since it is possible to be also found in close areas, which increases the error.

Table 1: Table with the results of the classification of the 3000 first images of the New College Data Set with the Viola-Jones and SAMME based scene recognition method, for $T = 5$ features.

| Alg. | Success | Neighb. | Error | Unass. |
|---|---|---|---|---|
| V&J | 64.8% | 15.0% | 11.5% | 8.5% |
| SAMME | 76.8% | 9.4% | 4.7% | 8.9% |

During the experimentation phase we encountered several problems. First, we realized that due to the limited number of images that could be used for training (images from the first round), it was possible to find features with error 0. The AdaBoost algorithm is not designed for features with error 0 (the feature is not weak, if it not fail classifying) so its $\alpha = \infty$, and only matching that feature the algorithm already considered that the image belongs to the node. This is not an easy problem to solve. First, we label the map as precise as possible trying to reduce the problem. We have also modified the algorithm to obtain in any case not an infinity $\alpha$.

During the experimentation, we tested two versions of AdaBoost. The first, proposed in paper (Romero and Cazorla, 2012a), is based on the Viola-Jones proposal (Viola and Jones, 2004) where the problem is considered as a multiple two-class problem. The second version corresponds to the scene recognition algorithm based on SAMME, which consider the problem from the perspective of multiple classes. The results can be seen in Table 1, where the first column are the version of the algorithms, the second one are the images well classified, "Neighb.' column are the number of images that have been classified as images belonging to a neighbor node from which it really belongs, "Error" are the misclassified images, and the last column is the number of images that are not classified in any node. The results have been obtained for $T = 5$. As we can see, with only 5 features (scene recognition is therefore very fast, allowing its use in real-time applications) the Viola-Jones scene recognition is able to classify approximately 79.93% of images (including success and neighbor columns), the error is 11.5% and the unassigned images are 8.57% of the total. However, using the new proposed SAMME scene recognition, the algorithm classifies approximately 86.33% of images, with an error of 4.73% and 8.9% of unassigned images. As we can see, although the unassigned images has increased, the number of images well classified (successes and neighbor images) have increased while the errors have decreased significantly.

Table 2: Table with the results of the classification of the 3000 first images of the New College Data Set with based scene recognition method, for $T = 5$ and $T = 50$ features.

| $T$ | Success | Neighb. | Error | Unass. |
|---|---|---|---|---|
| 5 | 76.8% | 9.4% | 4.7% | 8.9% |
| 50 | 83.9% | 9.5% | 6.2% | 0.2% |

Furthermore, we found that an increase in the number of features in the algorithm based on the Viola-Jones proposal means a decrease in the number of images well classified. Conversely, if we increase the number of features in the proposal with SAMME, the number of images well classified increases while the error does not increase significantly. Table 2 shows a comparison between the results obtained with $T = 5$ and $T = 50$ and the SAMME based scene recognition algorithm. As we can see, the success with $T = 50$ is 93.53%, while the error only increases to 6.23% and percentage of the unclassified images down to 0.2%.

Observing the partial results of each node separately, we have noticed that images added to its neighboring nodes are those at the borders between nodes. In addition, the images misclassified are mostly located on nodes belonging to the tunnel or its entrance and exit. As stated above, the images of the tunnel are very dark, with few features, so that the selected features are not good enough for classification.

## 4 CONCLUSIONS

In this paper we have presented a scene recognition method based on the SAMME AdaBoost algorithm using invariant visual features. The method gets good results, although, as noted, has several limitations. Once trained the method can be applied in real-time tasks, since it takes less than half a second to process one image (MSER+SIFT extraction and the comparisons with all nodes).

During the experimentation phase we have compare the SAMME based scene recognition algorithm with an adaptation of the Viola-Jones algorithm. As we have seen, the results obtained with the SAMME algorithm have been significantly higher than those obtained using the adaptation of the Viola-Jones algorithm.

The difficulties encountered during experimentation have focused on cases where the images are not good enough (low illumination, where it is difficult to extract features) and also in the manual classification of map images. Therefore, we should further study the behavior of the algorithm with other visual features (to be possible to withstand changes in illumination). And also raised the possibility of testing several different types of maps, to see if they improve the outcome in those nodes that are too big or small.

As future work, we intend to conduct further study on the behavior of the algorithm depending on the selected feature detector/descriptor (MSER, SIFT, SURF, Harris Affine, Hessian Affine, etc). Furthermore, we intend to merge an unsupervised topological mapping method and the AdaBoost, so that the positive and negative examples for the AdaBoost method will be provided to the topological mapping and the AdaBoost will return the features that should be used to find loop-closures.

## ACKNOWLEDGEMENTS

This work has been supported by project DPI2009-07144 from Ministerio de EducaciÃşn y Ciencia (Spain) and GRE10- 35 from Universidad de Alicante (Spain).

## REFERENCES

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28(2):337–374.

Hu, L., Jiang, S. Q. A., Huang, Q. M., and Gao, W. (2008). People re-detection using adaboost with sift and color correlogram. In *International Conference on Image Processing*, pages 1348–1351.

Liu, M., Scaramuzza, D., Pradalier, C., Siegwart, R., and Chen, Q. (2009). Scene recognition with omnidirectional vision for topological map using lightweight adaptive descriptors. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 116–121. IEEE.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110.

Mozos, Ó. M. and Burgard, W. (2006). Supervised learning of topological maps using semantic information extracted from range data. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 2772–2777.

Pajdla, T., Urban, M., Chum, O., and Matas, J. Robust wide baseline stereo from maximally stable extremal regions.

Romero, A. and Cazorla, M. (2012a). Learning topological SLAM using visual information. In *CCIA*, pages 151–159.

Romero, A. and Cazorla, M. (2012b). Topological visual mapping in robotics. *Cognitive Processing*, 13:305–308.

Smith, M., Baldwin, I., Churchill, W., Paul, R., and Newman, P. (2009). The new college vision and laser data set. *International Journal of Computer Vision*, 28(5):595–599.

Viola, P. A. and Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154.

Wu, J., Christensen, H. I., and Rehg, J. M. (2009). Visual place categorization: Problem, dataset, and algorithm. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 4763–4770. IEEE.

Yin, X. and Xie, M. (2007). Hand posture segmentation, recognition and application for human-robot interaction. In *International Conference on Advanced Robotics*.

Zhu, J., Rosset, S., Zou, H., and Hastie, T. (2005). Multi-class adaboost.