Knowledge-based Autonomous Dynamic Colour Calibration

Daniel Cameron, Nick Barnes

Department of Computer Science and Software Engineering, University of Melbourne
camerond@students.cs.mu.oz.au, nmb@cs.mu.oz.au

Abstract. Colour labeling is critical to the real-time performance of colour-based vision systems and is used for low-level vision by most RoboCup 2002 physically based teams. Unfortunately, colour labeling is sensitive to changes in illumination and manual calibration is both time consuming and error prone.
In this paper, we present KADC, a robust method for Knowledge-based Autonomous Dynamic Colour Calibration. By utilising the known geometry of the environment, landmarks are identified independent of colour classifications. Colour information from these landmarks is used to construct colour clusters of arbitrary shape. Clusters are dynamically updated through actions and by the use of a similarity metric, the Earth Mover's Distance (EMD). We apply KADC to the RoboCup Legged League, generating a colourtable purely from geometrical knowledge of the environment and dynamically update this colortable to compensate for non-uniform changes in lighting conditions.

1 Introduction

Image segmentation based on colour labeling is used extensively in real-time vision. Colour labeling requires colourspace to be partitioned into a set of colour classes, each corresponding to a symbolic colour such as 'blue' or 'green'. Each pixel is labeled with a symbolic colour according to location in colourspace and from this, objects of a distinct colour can be efficiently and reliably identified. For optimal effectiveness, colour labeling requires distinctive colour partitions that are not generally available in real world systems. In such circumstances, colour labeling can still serve as a useful component of multi-modal systems.

RoboCup playing environments have been constructed such that landmarks can be easily identified by their distinct colour scheme. By constructing a mapping between colourspace and colour classes of interest such as green (field), yellow (goal & beacons) and orange (ball), landmarks can be tracked in real-time. Unfortunately, this mapping degrades rapidly with any significant change in lighting conditions. As a result, RoboCup competitions are held under controlled lighting conditions designed to minimise colour misclassification. Colour classification poses a significant hurdle for the evolution of RoboCup to a more realistic environment. The lack of robust autonomous dynamic colour classification restricts RoboCup venues, requires an idealised playing environment, and contributes to long setup times.

In Robocup, determining which colourspace coordinates map to which colour classes has been commonly achieved through linear thresholding. The simplest form of this is achieved by thresholding each colour component, either in the native camera colourspace [1], or in a transformed colourspace [2]. Unfortunatly, colour classes may not correspond to the rectangular boxes in colourspace defined by these thresholds. To reduce misclassification, some RoboCup teams have used learning algorithms such as neural networks [3] and decision trees [3,4] to semi-autonomously generate arbitrary concave colour classes. Lookup tables in colourspace allow for the most accurate representation of colour classes as they handle arbitrary distributions. As with [1], colourspace fidelity is reduced due to the prohibitive cost of storing lookup values for the entire colourspace.

Current autonomous calibration techniques vary widely in their method of calibration and domain of application. Austermeier *et al.*, [5] presents a calibration technique that compensates for colour distortions from a reference illumination. Self-organising feature maps are used to compute a vector field for transforming pixels to their original colour under the reference illumination. This method is computationally expensive and requires a physical colour calibration chart to be available under all conditions to be calibrated for. Legenstein *et al.*, [6] presents a less powerful method suited for finding coloured objects that calibrates from a set of coloured strips carried by the robot.

Mayer *et al.,* [7] presents a method of semi-autonomous colour self-calibration using Mid-Sized League robots. The robot collects sample images from the known environment using colour classifications generated for previous lighting conditions. A filter to improve colour consistency is applied to these images which then undergoe k-means clustering. From each of these clusters, HSV thresholds are found that determine each colour class. These colour classes are then manually mapped to symbolic colours.

In this paper, Section 2 presents an overview of the system developed. The system is composed of three components: actions which allow for the rejection of landmarks that do not appear in plausible locations, colour independant landmark identification to eliminate classification divergence due to feedback, and a probabilistic colour clustering model which is formally defined in in Section 3. Section 4 outlines the Earth Mover's Distance (EMD), a metric used in image databases [8] adapted for use as a cluster similarity metric. Important implementation considerations as well as Legged League specific details are given in Section 5, with experimental results in Section 6.

The probabilistic cluster representation used makes no *a priori* assumptions about the distribution of the cluster in colourspace. Notably, concave and disjoint clusters can be represented without loss of information.

The approach we take considers a broader problem in colour calibration than most papers. Rather than model uniform lighting changes across the environment, we also consider non-uniform shifts in colourspace. If a light fails at one end of the field (as occurred during the 2002 Legged League competition) the illumination of the yellow landmarks drops, whilest the blue remain constant, and the pink is stretched in colourspace. Further, we consider the problem where the colour of landmarks is not known at all prior to calibration. Our results demonstrate the learning of landmark colours given no a priori information other than the geometry of the environment.

2 System Overview

In order to generate colour classifications, landmarks must be identified and the colour information from the landmarks used to generate the classifications. By taking actions based on geometric knowledge, these landmarks can be identified with greater certainty than a given set of random images. The system presented in this paper can be broadly decomposed into three components: actions, landmark identification, and colour classification.
Actions: A key difference between robot vision and computer vision in general is the ability to take actions. By using knowledge of the environment, actions can be used to increase the probability of particular landmarks being visible. By only considering landmarks when they are expected to be visible, false positives can be reduced, leading to more robust classification. By removing human interaction the overhead of classification is reduced such that it is practical to recalibrate extensively. This calibration can be performed dynamically during a game in parallel with existing systems.
Landmark Identification: Once the robot is positioned such that a landmark is expected to be visible, the image is segmented into 4-connected regions of self-similar colours using a segmentation algorithm independent of colour classifications (e.g., edges / regions). A landmark is composed of one or more sections of a single colour. Each section corresponds to a 4-connected region in the image. To identify a landmark, sets of regions are evaluated as to the probability that they correspond to the landmark. Spatial constraints known from the geometry of the environment (such as goals immediately above the field or beacon sections of similar size and vertically aligned) are used to evaluate the probability of a set of regions corresponding to a landmark.
Colour Classification: Once a landmark has been identified, the colour information from each section is used to construct a candidate cluster for each section. EMD is used to determine whether candidate clusters for the landmarks are sufficiently close to the current clusters of the colours associated with a known colour scheme (such as the current blue cluster and current yellow cluster for the goals in the Legged League). If a match is found, the current cluster is updated using the candidate cluster. For all updates, the probability that the region set corresponds to the landmark determines the magnitude of the update.

3 Clustering Framework

Cluster Representation: To handle the uncertainty present within classifications, cluster are represented within a Bayesian inference framework. Occupancy grids [9] are used in mobile robotics to model obstacles. Here we apply occupancy grids over colourspace to form the basis of clusters. An occupancy grid is used for each colour to map the distribution of that colour in colourspace. Colourspace is trivially divided into cells, each mapping to a distinct colourspace coordinate. Each cell C has an associated state variable $s(C)$ representing whether that cell maps to the colour $s(C) = \mathrm{COL}$, or to non-colour $s(C) = \mathrm{NON}$. These two discrete states are mutually exclusive and exhaustive thus the probability of a cell being in either state sums to 1 $P[s(C) = \mathrm{COL}] + P[s(C) = \mathrm{NON}] = 1$. Since $P[s(C) = \mathrm{NON}] = 1 - P[s(C) = \mathrm{COL}]$, the probability mass function $P[s(C) = \mathrm{COL}]$ defined over all cells defines the colour cluster for a given symbolic colour.

If no information is available, non-informative priors are used for initialisation:

$$P[s(C) = \mathrm{COL}] = P[s(C) = \mathrm{NON}] = 0.5 \tag{1}$$

Cluster Updating Since the probability of each cell mapping to a symbolic colour is treated as an independent random variable, we can use Bayes theorem to calculate the change in probability of cell C due to the classification M:

$$P[s(C) = \mathrm{COL}] \leftarrow \frac{P[M \mid s(C) = \mathrm{COL}]}{P[M]} P[s(C) = \mathrm{COL}], \tag{2}$$

Similarly, Bayes theorem also applies to NON:

$$P[s(C) = \mathrm{NON}] \leftarrow \frac{P[M \mid s(C) = \mathrm{NON}]}{P[M]} P[s(C) = \mathrm{NON}], \tag{3}$$

where

$$P[M] = \begin{array}{c} P[M \mid s(C) = \mathrm{COL}] \cdot P[s(C) = \mathrm{COL}] \\ +P[M \mid s(C) = \mathrm{NON}] \cdot P[s(C) = \mathrm{NON}]. \end{array} \tag{4}$$

Equations (2) and (3) can be combined and rewritten as:

$$\frac{P[s(C) = \mathrm{COL}]}{P[s(C) = \mathrm{NON}]} \leftarrow \frac{P[M \mid s(C) = \mathrm{COL}] \cdot P[s(C) = \mathrm{COL}]}{P[M \mid s(C) = \mathrm{NON}] \cdot P[s(C) = \mathrm{NON}]}. \tag{5}$$

For efficiency reasons, the log-likelihood of the cell probability is stored. The likelihood of a hypothesis H given the probability of that hypothesis $P(H)$ is:

$$L(H) = \frac{P(H)}{P(\neg H)}, \tag{6}$$

taking the log of this results in:

$$\log L(H) = \log\left(\frac{P(H)}{P(\neg H)}\right). \tag{7}$$

Expressed in terms of likelihoods, (5) becomes:

$$L[s(C) = \mathrm{COL}] \leftarrow L[M \mid s(C) = \mathrm{COL}] \cdot L[s(C) = \mathrm{COL}], \tag{8}$$

taking the log of both sides gives:

$$\log L[s(C) = \mathrm{COL}] \leftarrow \log L[M \mid s(C) = \mathrm{COL}] + \log L[s(C) = \mathrm{COL}]. \quad (9)$$

Which yields a simple update strategy: the cluster value $\log L[M \mid s(C) = \mathrm{COL}]$ due to M is added to the previous cluster value for each cell C.

Classification Representation: For each classification, a multiset of colourspace coordinates (one for each pixel in the landmark section) representing the colour information of the landmark is used to generate a new cluster which is subsequently merged with the appropiate symbolic colour cluster. Initially, the new cluster is uninformative with $\log L[s(C) = \mathrm{COL}] = 0$ for all cells C (corresponding to (1)). For each element of the multiset, the corresponding cell is updated as per (9).

Unlike sensors such as laser range-finders used in robot navigation, accurate sensor models are not available for landmark classification. In this paper we have modelled these probabilities as

$$L[M \mid s(C) = \mathrm{COL}] = \frac{f}{1-f} \exp\left(\frac{1}{A}\right) \quad (10)$$

where f is the fitness of the landmark and A is the image area of the associated region (both of which are defined below).

A colourspace coordinate close to a coordinate identified as part of a landmark section is more likely to correspond to the colour of the landmark. For example, if a pixel is identified as blue, pixels which are similar to the identified pixel are also likely to be blue. We model this by blurring updates over colourspace using Gaussian blur.

Cluster Decay: In uncontrolled environments, lighting conditions change over time and the clusters representing landmarks drift from their initial positions. To incorporate this into (9), a decay factor γ indicating the rate of depreciation of previous classifications has been introduced resulting in:

$$\log L[s(C) = \mathrm{COL}] \leftarrow \log L[M \mid s(C) = \mathrm{COL}] + \gamma \log L[s(C) = \mathrm{COL}] \quad (11)$$

Cluster Discretisation Since each cluster is treated independently, multiple symbolic colours can map to a single colourspace coordinate. If the colour labeling algorithm can utilise this ambiguity [10] then each cluster can be thresholded independently as given by (12).

$$S \in m(C) \leftrightarrow \log L_S[s(C) = \mathrm{COL}] > t_S \quad (12)$$

where $m(C)$ is the mapping from colourspace cell C to a set of symbolic colours, L_S is the likelihood for the cluster of symbolic colour S and t is the inclusion threshold of log-likelihood for S.

In many colour labeling applications, symbolic colours are mutually exclusive and each colourspace coordinate must be mapped to a unique symbolic colour.

In order to support such applications (which includes the Legged League), the cluster merging strategy is augmented to include a negative classification. That is, all clusters which are not being added to are subtracted from. For example, classifying a landmark as green implicitly classifies the associated candidate cluster as not red and not blue etc. Equation (9) could be augmented to modify all other clusters, however, this information is only needed during the discretisation process, a more efficient solution is to subtract when discretisation is required:

$$S \in m(C) \leftrightarrow \log L_S[s(C) = \mathrm{COL}] - \sum_{S_i \neq S} \log L_{S_i}[s(C) = \mathrm{COL}] > t_S \tag{13}$$

which guarantees mutual exclusion for all thresholds $t_S \geq 0$.

4 Cluster Similarity Metric - EMD

A cluster similarity metric is required in the cases where landmark identification cannot unambiguously determine which symbolic colour a region should be labeled and to prevent misclassified landmarks from being added to clusters. The former occurs in the Legged League for the goals and the beacons. These landmarks have identical geometry and differ only by colour and position.

This section briefly presents a overview of EMD (for a full description see [8]), and its application to dynamic colour classification. EMD was first proposed as a similarity metric between colour images and used for image retrieval and display in image databases. EMD derives it's name from the analogy that is the normalised minimum work required to fill a set of holes with earth by moving this earth from another set of mounds.

More formally, given a set of producers X each producing x_i units, a set of consumers Y each demanding y_j units, and a cost c_{ij} per unit of transportation from $i \in X$ to $j \in Y$ determined by the distance between i and j, the total cost of transportation is given by:

$$\sum_{i \in X} \sum_{j \in Y} c_{ij} f_{ij}, \tag{14}$$

where f_{ij} is the amount transported from i to j. EMD is defined as the normalised minimum total cost of transportation:

$$\mathrm{EMD}(X, Y) = \min \left(\frac{\sum_{i \in X} \sum_{j \in Y} c_{ij} f_{ij}}{\sum_{j \in Y} y_j} \right) \tag{15}$$

This minimisation problem is an instance of the classical transportation problem from linear programming, to which efficient solutions exist. Thus, EMD takes a two weighted sets of coordinates in some space, a ground distance in that coordinate space, and computes the minimum average distance required to move the first set to the second. The weight of each element of the set determines the capacity of that producer/consumer or in the earth moving scenario, the amount

of earth required to empty/fill the mound/hole. Rubner *et al.* [8] refers to these weighted sets of coordinates as 'signatures'.
Application to Dynamic Colour Classification: In the context of cluster similarity the signature of a cluster is determined directly from cell values. The signature of a cluster is composed of all colourspace coordinates with the weight of each coordinate equal to the normalised cell value. Cells with a value of zero may be omitted since they have no effect on the EMD. By normalising by the total sum of cell values, EMD is unaffected by cluster size and exists for all informative clusters. The ground distance used for calculating the cost of transportation is given by the straight-line (Euclidian) distance (L2 norm) between the coordinates in native colourspace. This ground distance does not correspond to the difference perceived between two colours but the colour distance as determined by the capture hardware. For applications requiring human interaction such as image databases, colourspaces that closely resemble the human perception of colour have been used [8].

5 Implementation

The system has been implemented on a Sony ERS-210 entertainment robot. A full size 2002 RoboCup Legged League field within a laboratory environment with partial natural lighting is used. The field comforms to Robocup 2002 Legged League requirements with exception that the white field lines are not present.
Cluster Compression: A naive implementation of probabilistic clustering using 24-bit colour, 32-bit floating-point precision and 10 colour classes requires $2^{24} \cdot 4 \cdot 10 = 640$Mb of storage. To reduce the memory footprint the least significant bits of each channels are ignored. By ignoring the lowest 4 bits of Y, 2 bits of U and V, the total memory required to store the clusters is reduced to 2.56Mb. Note that this corresponds to a reduced resolution occupancy grid over colourspace in which probabilistic formulation given in Section 3 also holds. To further compress clusters, clusters are tiled according to their most significant bits: 2 for Y and 3 for U and V. Colourspace thus is divided into 256 tiling blocks, each containing 256 units. Since clusters only occupy a small portion of colourspace, most blocks will have a value of zero in all cells and these are stored implicitly. It was found that on average, 8-16 blocks were occupied thus reducing the memory footprint from 2.56Mb to around 120Kb.

Calculating EMD directly from the cluster signature is computationally prohibitive so an approximation is used. For each tiling block, the centre of mass and total weight are used as the cluster signature. This information is cached for each tile for efficient EMD computation.
Actions: The action sequence undertaken by the robot attempts to maximise the probability of a landmark being visible when that landmark is being considered for cluster updating. The following sequence of actions was used for classification:

1. walk back 0.5m, look down
2. look up, turn around until a goal visible

3. walk to 1m from goal, look left, look right
4. turn around, goto step 2.

Initially, the robot must be on the field, when the robot looks down, the only landmark present will be the field, except is when the robot is at the edge of the field facing the white barrier. (1) reduces the probability of seeing the white barrier. Using (2), visible goals will be vertically centred in the image, simplify identification. Since there is one beacon either side of the goal, beacons should be visible and bounds can be placed not only on the expected image location but also on the expected size after (3). As the robot cycles between the goals, and its localisation converges to a line running between the two goals and the probability of finding beacons at the precalculated angles continually increases.
Landmark Identification: All landmarks in the Four Legged League are composed of one, or in the case of beacons, two homogeneous sections, each of a single colour. Since colour within a section varies only due to lighting conditions, and the colour of adjacent sections are widely separated in colourspace, both edge-based and region-based image segmentation techniques should be able to identify the regions of the image that correspond to landmark sections robustly. For this paper, an edge-based approach based on Canny edge detection [11] has been used. Canny edge detection is performed on each of the three input channels separately and the result combined in a boolean OR operation with an edge being present if it is present in at least one input channel with the result run-length encoded for efficiency. A region merging algorithm similar to [1] is used to identify all the 4-connected regions of non-edge in the image.
Landmark Fitness: Since the robot pose is known from the actions taken, simple heuristics can be used for landmark identification. The green field must occupy most of the lower part of the image. The goals must be vertically centred and should be twice as wide as they are high. Beacon sections must be of similar size and aligned vertically. Beacons and goals should be of the estimated size and approximately convex (the convex hull area is compared to the actual area).

6 Experimental Results

Static Illumination: To highlight the differences between calibration techniques, three colour classifications were generated. The first classification was generated on-board, updating only for landmarks that expected to be visible. The second was generated on a PC by transmitting images over a wireless LAN. Unlike the on-board classification, the PC-based autonomous classification updated all clusters without regard to expected image contents. The third classification was generated manually from the transmitted images. Due to limited bandwidth, the on-board autonomous classification was able to process more images and hence the image sequence used for this classification differs from the image set used for the autonomous PC-based classification and the manual classification.

To evaluate the three different methods, a second set of images was taken with the robot manually positioned in front of each landmark. These images served as a test set from which a reference classification was established by manually labelling regions of each image with appropriate colours. Misclassification rates were determined by comparing the classifications generated to the reference classification for each image in the test set. Due to the ambiguity in determining which edge pixels are considered to correspond to a landmark, the reference classification was conservative in its classification of landmarks and any false positives connected to the reference classification were assumed to correspond to the landmark.

	Manual	AIBO	PC
Beacon - Blue	1.0	2.2	1.1
Blue Beacon - Pink	0.3	0.1	0.6
Beacon - Yellow	9.4	3.3	4.2
Yellow Beacon - Pink	2.3	6.2	4.1
Goal - Blue	1.3	1.5	0.4
Goal - Yellow	8.6	6.1	4.0
Field - Green	2.0	0.3	2.5

Table 1. Misclassification rate (%) after self-calibration under constant illumination.

As can be seen in Table 1, all three classification techniques generated classifications of comparable quality. The high misclassification rate associated with the yellow clusters can be partially attributed to the small sample size as the robot spent most of its time near the blue goal and beacons.

Dynamic Illumination: A second experiment was conducted to evaluate the sensitivity to changes in illumination of the technique presented in this paper. For this experiment, four beacons were placed next to the blue goal as shown in Figure 1. Initially, the room was illuminated by two pairs of floodlights, and standard office neon lighting with the illumination gradually decreased to zero lighting, then increased to full lighting at a faster rate.

The first row of images shown in Figure 1 demonstrates the classifications after one floodlight is turned off. The manual calibration shown on the right exhibits some degree of degradation in classification of the blue goal. Note that since the autonomous classification only classifies the blue beacon (since no yellow goal has been seen hence the pink-yellow and pink-green beacons cannot be disambiguated), the pink section of the yellow beacon in the left of the image has been misclassified due to differences in illumination between the beacons.

In the second to fourth rows, illumination continually decreases. The autonomous classification shows little change whereas the static classfication completely misclassifies. The fourth row displays an instance of misclassification in the left yellow beacon due to poor image segmentation under the darker lighting conditions. The edge between the pink and the green was incomplete in previous frames and as such, the region classified as field also included the pink area of the beacon. Note however that this misclassification was rectified in subsequent frames and by the fifth row, the classification returned to pink.

To correctly re-identify landmarks under modified illumination, the change in illumination must be small enough that there is no ambiguity between the landmark under the new illumination, and other landmarks of the same shape of different colour. The fifth row shows classifications after an abrupt increase in illumination. Note that unlike the field and goal, the classification of the blue beacon was unable to recover from the abrupt change.

7 Conclusion

In this paper, we presented KADC, a new method for Knowledge-based Autonomous Dynamic Colour Calibration. This method and algorithm facilitates the generation of new colour classifications, and dynamically updates existing colour classifications under changing lighting conditions given certain assumptions. Specifically, that a method is available for segmenting the regions independent of colour classification, that we have knowledge of where important features might occur in the environment, what they might look like, and a robot which is able to perform actions to look for these features. We have implemented this algorithm on an embedded robot platform (Sony AIBO), and have used it to generate partial colour tables for the Legged League competition given no apriori colour information. The method was able to generate colour classifications that were comparible to hand segmented clusters under constant illumination. Further, under changing lighting conditions, the technique presented is able to dynamically update classifications to maintain a similar quality of classification. Under these changing conditions hand generated clusters have high error rates. This is a significant development for RoboCup, where in the physical leagues, teams have issues of classification robustness, are only able to compete under controlled lighting conditions, and require long setup times.

The use of a probabilistic interpretation of colour classes allows for uncertainty in classifications to be modeled explicitly and for colour classes to be represented without loss of information. The application of EMD to cluster similarity allows for the comparison of colour cluster and hence a measure of similarity between arbitrary image regions.

The techniques presented are applicable to colour-based vision applications in general. The method of calibration presented represents a significant improvement in the robustness of colour classification and allows colour labeling to be used in a more general class of problems with dynamic non-uniform illumination.

References

1. Bruce, J., Balch, T., Veloso, M.: Fast and inexpensive color image segmentation for interactive robots. Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '00) **3** (2000) 2061–2066
2. Oda, K., Kato, T., Ishimura, T., Katsumi, Y.: The kyushu united team in the four legged robot league. In: Proceedings of RoboCup 2002, The 2002 International RoboCup Symposium. (In Press)

3. Akm, H.L., Pavlova, P., Yildiz, O.T.: Ceberus 2002. In: Proceedings of RoboCup 2002, The 2002 International RoboCup Symposium. (In Press)
4. Cayouette, F., Sud, D., Patel, K., Sarikaya, D., Cooperstock, J.: Mcgill reddogs 2002 team description report. In: Proceedings of RoboCup 2002, The 2002 International RoboCup Symposium. (In Press)
5. Austermeier, H., Hartmann, G., Hilker, R.: Color-calibration of a robot vision system using self-organizing feature maps. Artificial Neural Networks - ICANN 96. 1996 International Conference Proceedings (1996) 257–62
6. Legenstein, D., Vincze, M., Chroust, S.: Finding colored objects under different illumination conditions in robotic applications. In: Proceedings of SPIE Vol. 4197 Intelligent Robots and Computer Vision XIX: Algorithms, Techniques, and Active Vision. (2000)
7. Mayer, G., Utz, H., Kraetzschmar, G.K.: Towards autonomous vision self-calibration for soccer robots. IEEE/RSJ International Conference on Intelligent Robots and Systems (2002) 214–219
8. Rubner, Y., Tomasi, C., Guibas, L.: A metric for distributions with applications to image databases. IEEE International Conference on Computer Vision (1998) 59–66
9. Elfes, A.: Occupancy grids: a stochastic spatial representation for active robot perception. In: Proceedings of the Sixth Conference of Uncertainty in AI. (1990) 60–70
10. Akm, H.L., Pavlova, P., Yildiz, O.T.: Ceberus 2002. In: Proceedings of RoboCup 2002, The 2002 International RoboCup Symposium. (In Press)
11. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence **PAMI-8(6)** (1986) 679–698

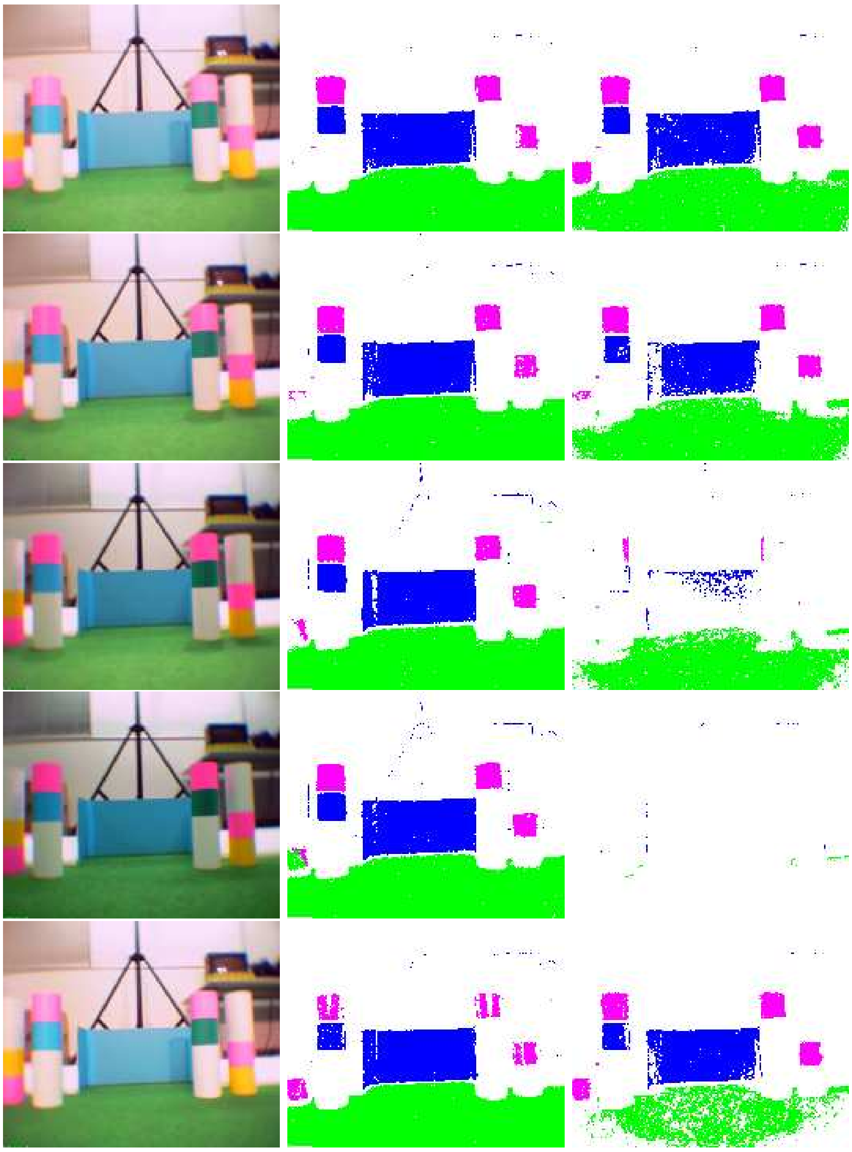

Fig. 1. Lighting conditions are dynamically changed during the experiment. On the left is the actual image, the dynamically calibrated classification in the middle, and a static classification done at the start of the experiment on the right.