On the reduction of raw image data in the context of the appearance-based approach to robot navigation

Martin Hülse, Frédéric Labrosse and Mark Lee

***Abstract*— The appearance-based approach towards robot navigation is based on a pixel-wise comparison of images. Recent research has shown that the Euclidean distance in image space provides a robust method for robot homing, navigation along routes and topological mapping. The objective of this paper is to investigate how image data can be reduced in order to minimise the computational cost for the image distance calculation without loosing the robustness of the method. A simple 1-D scenario is used to test three different types of reduction methods: one focuses on specific and predefined image regions, one uses fractal sets, while the last is based on a stochastic process. We show that with less than 10% of the data a similar performance can be achieved with the stochastic method, which is then used on a real case study (the visual compass) to assess its performance in a real situation.**

I. INTRODUCTION

The *appearance-based* approach towards visual navigation for autonomous robots has been shown to be a robust method for indoor and outdoor applications [5]. In particular, it was shown that a simple pixel-wise comparison between images taken along a route (way-images) and the current image is enough to create topological maps [9] and to make a robot navigate along routes composed of a succession of way-images [6] without any topographical information. More specifically, the methods were based on the Euclidean distance between images $\mathcal{I}_i$ and $\mathcal{I}_j$:

$$d(\mathcal{I}_i, \mathcal{I}_j) = \sqrt{\sum_{k=1}^{w \times h} \sum_{l=1}^{c} \left(\mathcal{I}_i(k,l) - \mathcal{I}_j(k,l)\right)^2}, \qquad (1)$$

where $\mathcal{I}_i(k,l)$ represents the l-th color component of pixel k. In consequence, robot navigation tasks can be performed using raw visual input and don't need to involve any filters, feature detection processes or landmarks detection/recognition.

A similar approach has been used to visually track objects specified only by their appearance, i.e., an image of them. The Euclidean distance was then used to locate in live images a geometric transformation of the appearance that takes into account changes in view point. Two different applications of the method were proposed in [8], where the systems ran in real-time using full images.

Nevertheless, autonomous robot systems usually have limited CPU speed and memory. Both limitations suggest that an application of the appearance-based approach does fit best autonomous systems of high robot-environment interaction dynamics, if data reduction can be involved. By data reduction, in contrast to compression, we mean that the distance measure between two images involves far less pixels per image than originally provided by the camera.

Former studies have indicated that data reduction techniques create better error-surfaces if sampling covers the whole image data rather than using pre-selected and fixed image regions [2]. This is in fact a not surprising conclusion (that we confirm ourselves later) given that the robustness of appearance-based methods is in part due to the global nature of Eq. (1). Indeed, because all the pixels contribute to the distance between images, small, localised changes in appearance don't affect dramatically the comparison of appearances. Any departure from the global nature of the method can in fact create artifacts in the image comparison that in turn will change the behaviour of the system ([5] gives a good example of that, see Section VI).

In this paper we propose, compare and evaluate three methods of data reduction and present their suitability for the appearance-based approach. Although this approach is usually applied to panoramic images and robot movements in two dimensions or more, our initial investigation deals only with a one-dimensional scenario. However, the reader will see that these experiments lead to interesting insights, which can be directly applied to the 2-dimensional standard domain.

In Section II we introduce our experimental setup, which performs the distance measures between different images in one-dimension. Section III defines the three methods for data reduction. The first is based on selected image-regions, the second is based on a stochastic approach, while the last uses image coverages of fractal dimensions. Section IV presents results of applying the three coverages to a specific dataset. In Section V, the stochastic method is used in a real application, that of estimating the orientation of a robot using only the succession of images along its path [6]. In Section VI we evaluate our results from a more general viewpoint, leading us to suggestions for the classical two-dimensional domain of the appearance-based approach.

II. A 1-DIMENSIONAL TEST SCENARIO

Our test scenario involves a pan-tilt-verge system, which can be equipped with two cameras. In this context, we used only one camera and the corresponding verge-axis. A sample of 51 images was recorded, equally distributed within the interval $(-\frac{\pi}{4}; \frac{\pi}{4})$. The camera has a horizontal and vertical field of view of respectively $94°$ and $69°$ (see Fig. 1). Thus, the sampled images cover a horizontal visual field of $180°$. The original 1032×778 resolution of the images was reduced to 729×729 in order to apply all three data reduction

Dept. Computer Science, Aberystwyth University, Penglais, SY23 3DB, Wales, UK [msh,ffl,mhl]@aber.ac.uk

methods directly for the same data set. Indeed, the fractal reduction fits best with this dimension and for the other methods the dimension is not essential. Furthermore, the use of raw image data means, in our scenario, that we directly operate on the delivered Bayer pattern image data without creating the RGB color images before image distances are calculated (Fig. 1) [1], [11].

The image captured at position 0.0 was used as reference image $\mathcal{R}$. Each image $\mathcal{I}_i$ has now a specific distance $d(\mathcal{R}, \mathcal{I}_i)$ to this reference. This distance value, here a scalar, is essentially the only piece of information the whole appearance-based methods use. The solid curve in the diagram shown in Fig. 2 illustrates the typical evolution of the distance values while the camera starts at position $-\frac{\pi}{4}$ and moves towards position $\frac{\pi}{4}$, going through a zero distance at position 0.0, which is the position of the reference image $\mathcal{R}$.

The distance values are normalized by dividing them with the number of pixels involved in the image distance calculation. Hence, the values represented by the solid curve in the diagram of Fig. 2 are multiplied with $\frac{1}{729 \cdot 729}$, since all pixels were used. In contrast the second curve (dashed line) of the same diagram uses only 19,721 pixels ($\approx 3.7\%$) of the original image data. Despite this significant data reduction we have an unexpectedly good match with the curve where all pixels were used. The actual data reduction applied in the second sample is indicated by the bottom image in Fig. 2. The dark dots in this image, together creating the *mask* or *coverage*, indicate those pixels involved in the calculation of the distance. This coverage is based on a fractal structure, namely Sierpiński carpets [7]. Encouraged by this promising example, we have systematically investigated different methods of data reduction based on three distinct types of data coverage.

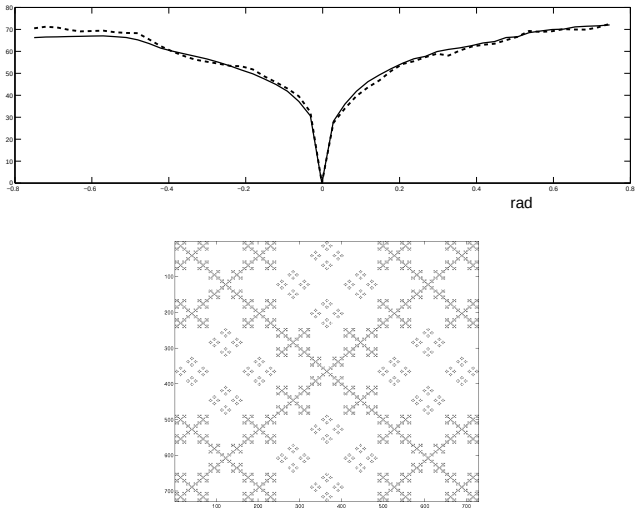

Fig. 2. Mask representing a specific coverage of raw image data (bottom). The black dots represent the pixels used for the calculation of image distance and cover only 3.7% of the whole image. The diagram (top) shows the evolution of image distance values operating on the reduced dataset (dashed line) in comparison to image distance values based on the whole dataset (solid line). The mean error and standard deviation of the error between these two graphs are 3.18 and 4.50, respectively.

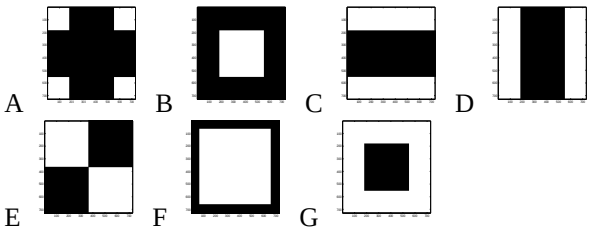

Fig. 3. Masks for the region-based image data reduction. Notice, each of these "small" images represents the coverage of a 729×729 image.

III. Data reduction approaches

We have investigated three different types of image data reduction. In the first method only specific regions in an image are considered. We call it *region-based* image coverage. The masks we applied to reduce the data are shown in Fig. 3. They provide reductions of 75%, 50%, and significantly less than 50% of the data. Black regions indicate the used pixels of the images of resolution 729×729.

The second image reduction method is based on a random process. Whether or not a given pixel is involved is randomly chosen based upon a given probability p and a uniformly distributed random process. In that way p obviously determines the amount of data used. Once an image coverage is determined in this manner, it is fixed and is applied for the computation of all distance values. In the following we call this method *probability-based* image coverage.

Fig. 1. Bayer pattern images for three camera positions: $-\frac{\pi}{4}$ (left), 0.0 (middle), and $\frac{\pi}{4}$ (right).

The last method lies qualitatively between the region and probability-based approaches. It is based on fractal sets, called Sierpiński carpets [7]. Sierpiński carpets result from an iterative process where a pattern is successively used to replace specific regions in an evolving pattern. In Fig. 4 two examples are shown, illustrating this process. One starts with a given pattern, here a square divided into 3×3 equal sub-squares. The sub-squares are labeled either black or white, which creates a specific pattern. At each iteration the original pattern is used to replace all the black labeled regions, by this pattern again. This leads to a finer partition of the originally given pattern. After the first iteration, when each black labeled square is replaced by the pattern, we have a square subdivided into $9 \times 9 = 81$ regions, instead of the 9 regions given in the original pattern. After 5 iterations we have a "fractal set" represented on a square, regularly

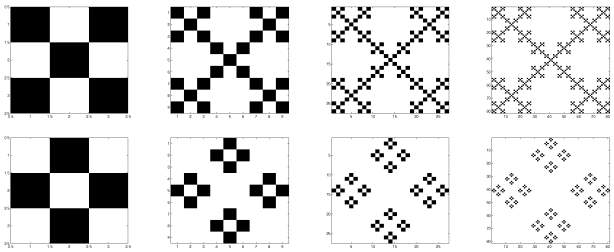

Fig. 4. Two examples of the first three iterations towards Sierpiński carpets. The first pattern (top) has a larger fractal dimension than the second and therefore the coverage is larger.

subdivided into 729×729 equal sub-squares. This pattern can directly be interpreted as a coverage or data reduction mask for our 729×729 image data.

Having a 3×3 sub-divided square as original structure, there are 2^9 possible masks generating different image coverages. For trivial reasons, we can exclude the two cases where all 9 sub-squares of the original pattern are labeled with the same color. Further more, we have selected out those initial patterns containing only 3 or less black labeled sub-squares. The remaining patterns will generate sets having a fractal dimension larger than 1, which is needed for masks covering a 2-dimensional space [7]. One has to notice that in consequence a mask, and therefore the resulting coverage, can have one of five different fractal dimensions. For our 3×3 initial patterns the fractal dimension $d_f(n)$ is defined as:

$$d_f(n) = \frac{log(n)}{log(3)},$$

where $n \in \mathbb{N}$ is the number of black labeled sub-squares in the initial pattern and per definition we have $4 \leq n \leq 8$.

IV. Experiments and results

In order to compare the different coverages we calculate the mean error $\overline{x}$ and standard derivation σ^2 between the two distance values resulting from the use of the reduced and the complete data set:

$$\overline{x} = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{d(f(\mathcal{R}, M), f(\mathcal{I}_i, M))}{card(M)} - \frac{d(\mathcal{R}, \mathcal{I}_i)}{w \times h} \right)^2,$$

where N is the number of images (here we have 51) and $f(\mathcal{I}, M)$ represents the reduced image data, if mask M is applied to image $\mathcal{I}$. The term $card(M)$ indicates the number of pixels of coverage M. The value of $card(M)$ is used for normalization. Therefore, an image distance without data reduction is normalized by the number of pixels of the original image (given as the product of width w and height h). In the presented scenario we have 729^2 as normalization factor for the original data set.

A. Region-Based Coverage

The results of the region-based data reduction are presented in Table I.

Not surprisingly the larger the region is, the smaller the error becomes. However, even in the best cases (masks A and B), the absolute errors have magnitudes of over 10% of the reference data (no reduction), which values range between 0 and approximately 65. On the other hand, we have significantly different error values for the three masks C, D, and E, although all of them use 50% of the data.

TABLE I
Error values of region based image coverage

coverage	$\overline{x}$	σ^2	rel. data (%)
A	7.7	7.6	≈ 75
B	4.7	4.0	≈ 75
C	63.3	64.3	≈ 50
D	47.2	38.7	≈ 50
E	102.8	92.2	≈ 50
F	37.4	51.2	≈ 35
G	42.6	39.0	≈ 25

These results tell us that the structure of the regions (form, placement and orientation in the image) has a big impact on the error, and not only the amount of data which is used. All in all, the region-based data reduction seems to be not a good choice, since suitable regions/structures can hardly be estimated a priori. Thus, the whole method is rather inapplicable, especially for autonomous robots in changing environments.

B. Fractal Coverage

As we have mentioned above, there are five different types of fractal coverage, which are distinct due to their fractal dimension d_f. It is important to mention that equal fractal dimension means that the corresponding coverage size (number of black labeled pixels) is equal too. Hence, even if the resulting structures of the masks look very different, the masks provide the same amount of data as long the fractal dimension is the same.

In consequence, we have five groups of masks with equal fractal dimensions and all the masks in one group provide the same number of pixels.

According to the definition above, we have 381 masks of fractal dimension in the range 1 to 2. From these 381 masks we have only tested 50, because the remaining masks can be transformed (by rotation or mirroring) into one of these 50 examples.

It turned out that the error values can be significantly different between two masks although they have equal fractal dimension (see Fig. 5 (top)). Hence, again it seems the structure of the initial patterns highly influence the quality of the data reduction.

Selecting the best mask for each fractal dimension, we obtain the errors given in Table II and shown in Fig. 5 (bottom). Considering the mean error values, we can again point out that a larger fractal dimension corresponds to less error. However, comparing the best fractal coverages with the region-based examples we can see much smaller absolute errors. For instance, if $n = 8$ we have a fractal coverage using almost 50% of the data, which is similar to masks C, D, and E from the region based method. In this case the mean

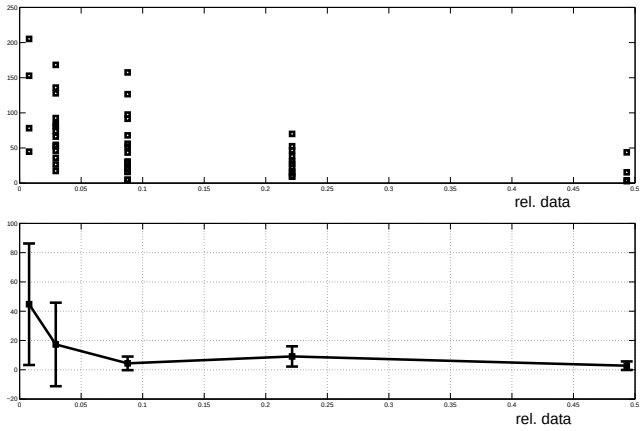

Fig. 5. Top diagram: mean error $\overline{x}$ over the amount of data for all 50 basic masks of fractal dimension d_f (see text for details). Bottom diagram: mean error and standard derivation of the best mask for each fractal dimension.

TABLE II

MEAN ERROR VALUES FOR THE COVERAGE OF FRACTAL DIMENSION, USING THE BEST EXAMPLES FOR EACH FRACTAL DIMENSION.

fractal dim. d $d_f(n) = \frac{log(n)}{log(3)}$	$\overline{x}$	σ^2	best $(\overline{x}; \sigma^2)$	rel. data (%)
$n = 4$	120.2	106.8	$(44.8; 41.5)$	≈ 0.8
$n = 5$	75.9	89.9	$(17.3; 28.5)$	≈ 2.9
$n = 6$	50.8	53.1	$(4.4; 4.7)$	≈ 8.8
$n = 7$	30.6	28.3	$(9.1; 6.9)$	≈ 22.1
$n = 8$	16.5	21.0	$(2.8; 2.9)$	≈ 49.3

error (16.5 ± 21.0) of the fractal coverage is significantly smaller than for D (47.2 ± 38.7), the best candidate of the region based method.

Again, we see that the structure of the fractal coverage, determined by the labeling of the initial patterns, highly influences the quality of the coverage. An a priori estimation, indicating which structure leads to reasonable small errors for a specific environment, does not exist.

However, fractal coverages operate on much less data (approximately 1–50%), which makes it hard to relate the overall result to that of the region based approach. This gap can be closed by the the probability based method described next.

C. Probability-Based Coverage

As mentioned above, in the probability based method the pixels involved in the image distance calculation are randomly selected. Once such a random mask is generated for a given probability p, it is fixed for the distance calculation of the whole sample. For each value of p we generated ten different masks and applied them independently for the given sample. Table III summarises the results for selected values of p. Notice that the mean value and standard deviation in this table refer to different qualities compared to the tables above. We now have the mean and deviation of the error values resulting from the 10 tests for fixed p, but each test running with a new, randomly generated mask. As it is shown in

TABLE III

ERROR VALUES OF THE PROBABILITY BASED IMAGE COVERAGE (10 RUNS PER GIVEN PROBABILITY p).

prob. p	mean	std. deviation	rel. data (%)
0.01	0.49	0.62	≈ 1.0
0.03	0.15	0.19	≈ 3.0
0.08	0.07	0.08	≈ 8.0
0.12	0.05	0.05	≈ 12.0
0.25	0.01	0.02	≈ 25.0
0.50	0.01	0.01	≈ 50.0

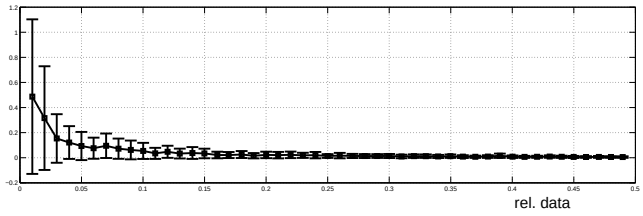

Fig. 6. Mean and standard deviation of the errors resulting from different probabilities p, while for each value of p we have 10 runs.

Fig. 6 we tested the whole interval for probabilities between 0.01 (1% of the pixels) and 0.5 (50% of the pixels). The results indicate that the probability based method delivers the best results, without any exception. Rather surprisingly, they generate best and almost perfect matchings using only 1% of the data. With $p = 0.01$, the errors vary only around ≈ 0.5. Comparing these variations with the absolute values of the image distances (ranging between 0 and 65) then we can say that for the appearance-based approach to robot navigation such small errors are rather irrelevant. Further more, for $p > 0.1$ we have, in effect, an error of zero. However, small errors can have a dramatic effect when accumulated, as it will be shown in the next Section V.

D. Summary

The overall picture of the three reduction methods is summarized in Fig. 7. The less structured our coverage is the better the matching with the original data becomes. The probability-based method, even with only 1% of the data (thin black line) provides a matching that none of the other methods can achieve.

V. CASE STUDY: THE VISUAL COMPASS

In order to test the theory presented above in a real situation, we ran an experiment previously published [5], reducing the images using the probability-based coverage.

The visual compass uses a property of panoramic images: they only present a horizontal shift of their pixels if the camera rotates around its optical axis. Therefore, the change in orientation between two images grabbed from the same position can be estimated by finding the best match between the two images, using the Euclidean distance as presented above, when "rotating" (column-wise shift) the second image. Such a match is also possible if the robot (and hence the camera) translates between the two images, albeit

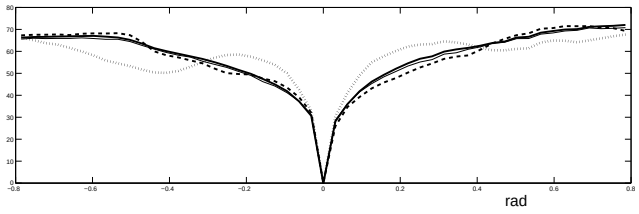

Fig. 7. Evolution of distance values for four different coverages. The bold solid line represents the distance values without data reduction (reference data). The doted line represents the region-based coverage applying mask G operating on 25% of the data (see Fig. 3). The dashed line indicates the distance values resulting from the fractal coverage using the best mask for $n = 6$ (8.8% of the data used). Finally, the thin solid black line (almost invisible due to its tiny difference to the reference data) represents data resulting from the probabilistic method for $p = 0.01$ (1% of the data).

with a match of a lower quality, depending, amongst other, on the amount of translation. By carefully choosing which images to consider from the sequence of grabbed images, one can accumulate local rotations to provide the orientation of the robot along its path relative to the orientation at the beginning. Details of the method are presented in [5].

The experiment was as follows. A robot carrying a panoramic camera was randomly driven in various environments, grabbing and saving to disk images as it went along, with ground truth orientation provided by a magnetic compass. Note that the visual compass can be run in real time (and has been used in other work as the only source of orientation information [6]). The saved images (reduced with the probability-based coverage) are then used to compute the orientation of the robot along its path, the orientation being compared to the ground truth for evaluation of the method. We present here results obtained for the dataset GRASS1 from [5]. The images are 360×45 and only the regions corresponding to the front and back of the robot are considered for the visual compass (because the side areas convey very little rotation information when the robot translates), only using a maximum of 5400 pixels out of the 16200 (two orders of magnitude less than in Section IV).

Fig. 8 shows the error between the ground truth data and the orientation computed by the visual compass for a number of values of p in the probability-based coverage. For $p = 1.0$, all the pixels are retained and the graph is a subset of the one presented in [5, Fig. 18]. It is clear that only from $p = 0.2$ significant differences start to appear, with an obvious degradation for $p = 0.05$.

One of the important aspects of the visual compass is that it provides an estimation of the orientation that drifts only very slowly. This property is desirable and we need to make sure that the partial coverage does not destroy it. Because the coverage is based on probabilities, we need to evaluate the performance of the visual compass for a number of coverages for each given value of p.

Figs. 9 to 12 show the error graphs for 10 different coverages corresponding to values of p equal to, respectively, 0.2, 0.3, 0.4 and 0.5. These used an average number of pixels, respectively of 1078, 1611, 2165 and 2719. Although

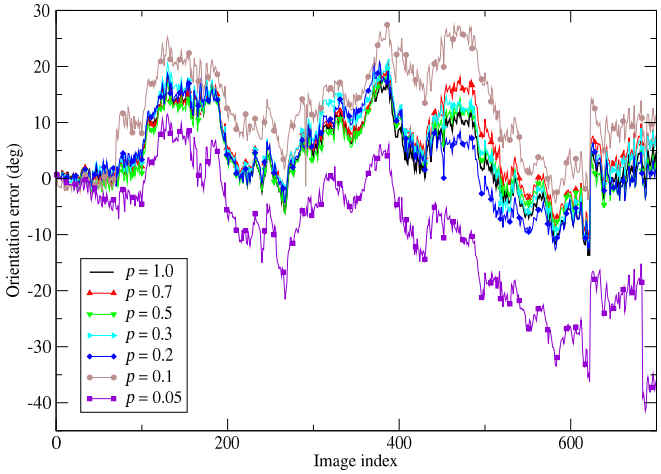

Fig. 8. Orientation error for various probabilities of the probability-based coverage.

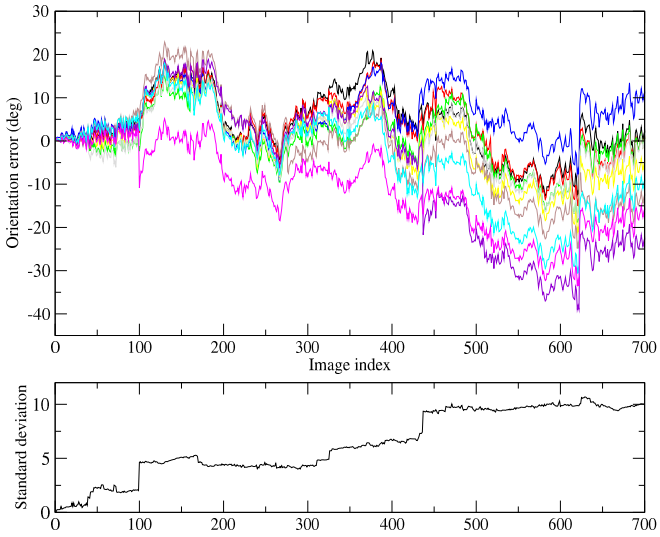

Fig. 9. Orientation error for 10 probability-based coverage with $p = 0.2$ and standard deviation of the error for each image in the sequence.

it is clear that with a probability of 0.2 the orientation drifts significantly (the standard deviation increases), the drift is reduced to very little when 50% of the pixels are kept.

VI. Discussion and Conclusion

We have presented three methods to select a sub-set of pixels to be used in image comparison, with various degrees of structure. The results show that in the absence of a priori information, the random scheme performs much better than the other methods. It is not unexpected that the region-based method would fail as the regions where chosen without taking into account the content of the images, which is usually not known and about which it is not desirable to make assumptions because of often changing robot environments.

It is more surprising for the fractal coverage to perform not as well as the random method. This is because the chosen fractal coverage in fact creates large areas without any pixels kept while others will have a large number of them, therefore

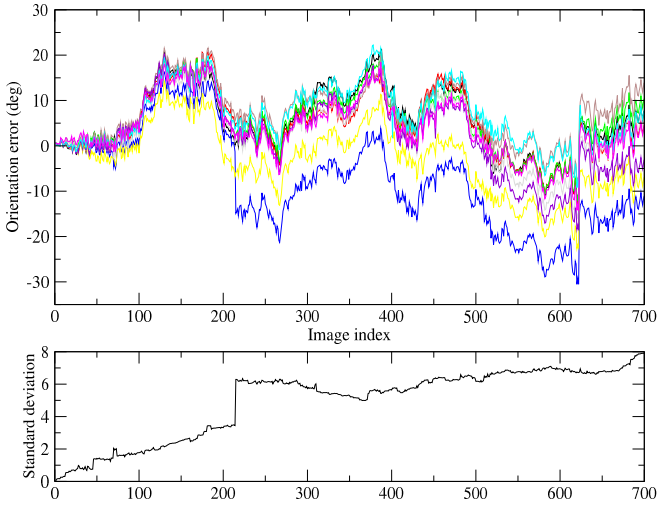

Fig. 10. Orientation error for 10 probability-based coverage with $p = 0.3$ and standard deviation of the error for each image in the sequence.

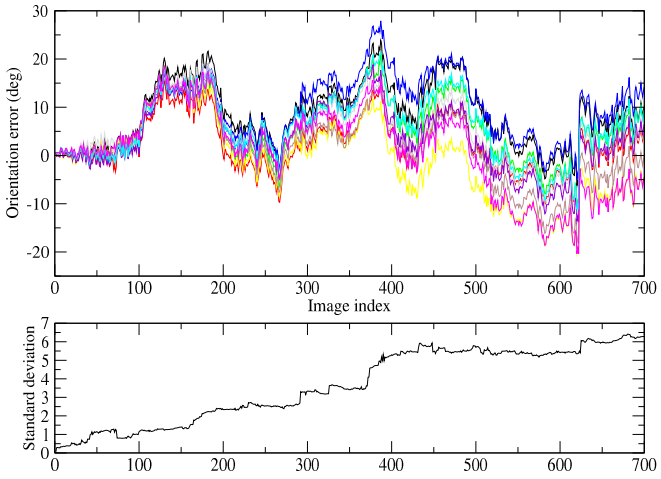

Fig. 11. Orientation error for 10 probability-based coverage with $p = 0.4$ and standard deviation of the error for each image in the sequence.

making these coverages not dissimilar from the region-based ones. It is possible that Peano, or similar, curves might perform better because of their more homogeneous coverage of the image plane.

Other schemes, based on straight lines randomly organized on the image plane have been tried and they perform well, similarly to the random scheme presented here [2]. However, we stress the fact that repetitive structures need to be randomly positioned by, e.g., adjusting the spacing. Indeed, if a regular structure is used, patterns such as the ones observed when re-sampling images to obtain sub-pixel accuracy in the matching would appear [10], [3]. In fact, without knowledge about the content of the images and extra processing, any scheme that uses a systematic coverage (such as the region-based, and to some extent the Sierpiński-based fractal, schemes proposed here) is bound to not be as successful as a random coverage. In fact, the visual compass does perform a selection: only the parts of the images corresponding to the front and back of the robot

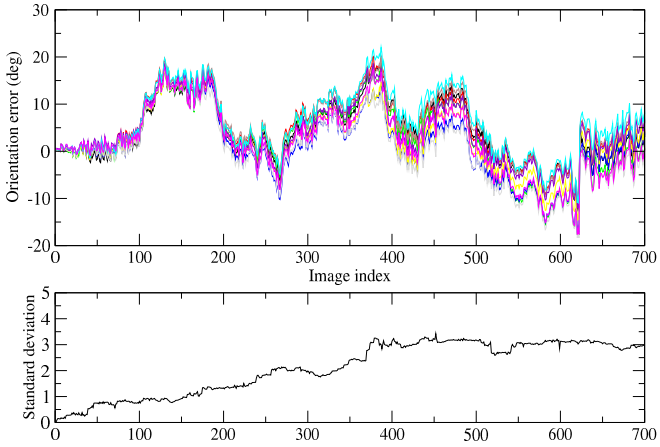

Fig. 12. Orientation error for 10 probability-based coverage with $p = 0.5$ and standard deviation of the error for each image in the sequence.

are used because they contain most of the robot's rotation information [5]. This could be viewed as a region-based scheme as presented here. However, the regions here do not correspond to image content but information content. Moreover, it was shown in [5] that this can lead to bad performance and a specific example was discussed at length. A car, significantly distinct from the robot's environment, moving relative to the robot was suddenly appearing in the effective field of view. This resulted in "pulling" or "pushing" the compass by either keeping the car out of the field of view or stabilising its position in the field of view.

We have applied the random coverage to the visual compass [5] and have shown that its performance does not degrade significantly with as few as 50% of the pixels. Notice that the visual compass was already performing an important data reduction by ignoring large areas of the images that contain little rotation information. The amount of pixels kept in fact corresponds to 17% of the original complete image. This is significantly more than the 1% mentioned in Section IV-C however. This is because the visual compass accumulates local estimations of changes in orientation, therefore accumulating errors, however small these might be.

The aim of the reduction of the number of pixels used was to reduce the computational complexity. Eq. (1) shows that the complexity of the distance measure is linear with the number of pixels considered. Moreover, previous publications have shown that such method can be used in real time (of the order of 10 frames per second) on low power computers (Pentium III at 800MHz). A reduction to 50% of the pixels would roughly double the frame rate (in fact not quite because there are other aspects involved in the computation). Alternatively, the resolution of the images could be doubled, therefore improving the performance of the system. It was indeed shown in [4] that increasing the resolution of the panoramic images was dramatically improving the visual compass, but this was discarded in [5] because of the computational cost of such an increase. This needs to be tested further.

Another avenue concerns the use of neural networks. Some success has been shown in tasks such as object recognition or robot navigation using images as input of neural networks which essentially perform pixel-wise comparisons. However, the neural networks could only use very low resolution images for performance reasons. We have shown that an alternative would be to use a selection of pixels from high resolution images.

The examples presented here only covered cases with one degree of freedom. However, there is no reason to believe that a similar behaviour will not be seen in cases with more degrees of freedom. In fact, the object tracking work presented in [8] uses a sub-sampling of the images in some cases: when the appearance is made larger, e.g., when the object to track is too large, then only a subset of the pixels in the corresponding area of the live image is used. This sub-sampling covers the whole area of interest and therefore is similar to our probability-based coverage. However, the number of pixels contained in the appearance is typically low (of the order of the hundreds) and any further reduction is likely to be detrimental. On the other hand, the proposed reduction schemes will certainly help with the computational complexity of the whole mapping and navigation system. This needs to be evaluated.

The important contribution of this paper is to show that reducing the amount of pixels is not detrimental to appearance-based methods. This is the case provided the reduction is performed appropriately. In cases where no information is available about the image content or the specifics of the task then a probability-based coverage should be the preferred method.

VII. Acknowledgments

This work was partly supported by the EC-FP7 projects IM-CLeVeR and ROSSI, and through UK EPSRC, grant EP/C516303/1.

References

[1] B. E. Bayer. Color imaging array. US Patent No. 3971065.

[2] H. Hosseini, M. Neal, and F. Labrosse. Error surface generation techniques for appearance-based stabilisation of an intelligent kite aerial photography platform (ikapp). *In: S. Ramamoorthy and G. M. Hayes (eds.) TAROS 2008, Edinburgh*, pages 163–170, 2008.

[3] J. Inglada, V. Muron, D. Pichard, and T. Feuvrier. Analysis of artifacts in subpixel remote sensing image registration. *IEEE Transactions on Geoscience and Remote Sensing*, 45(1):254–264, 2007.

[4] Frédéric Labrosse. Appearance-based heading estimation: the visual compass. Technical Report UWA-DCS-06-048, Computer Science Department, University of Wales, Aberystwyth, UK, 2006.

[5] Frédéric Labrosse. The visual compass: Performance and limitations of an appearance-based method. *Journal of Field Robotics*, 23(10):913–941, 2006.

[6] Frédéric Labrosse. Short and long-range visual navigation using warped panoramic images. *Robotics and Autonomous Systems*, 55(9):675–684, 2007.

[7] Benoît Mandelbrot. *The Fractal Geometry of Nature*. Freeman, San Francisco, 1982.

[8] James McIntyre, Andrew Church, and Frédéric Labrosse. Efficient image-based tracking of apparently changing moving targets. In *Proceedings of Towards Autonomous Robotic Systems*, pages 119–126, University of Ulster, UK, 2009.

[9] Mark Neal and Frédéric Labrosse. Rotation-invariant appearance based maps for robot navigation using an artificial immune network algorithm. In *Proceedings of the Congress on Evolutionary Computation*, volume 1, pages 863–870, Portland, Oregon, USA, 2004.

[10] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Interpolation artefacts in mutual information-based image registration. *Computer Vision and Understanding*, 77:211–232, 2000.

[11] C. Sakamoto, T. Nakanishi and T. Hase. Software pixel interpolation for digital still cameras suitable for a 32-bit mcu. *IEEE Transactions on Consumer Electronics*, 44(4), 1998.