A Region Adjacency Tree Approach to the Detection and Design of Fiducials

E. Costanza and J. Robinson

Department of Electronics, University of York, York, United Kingdom

Abstract

We report a topological approach to fiducial recognition for real-time applications. Independence from geometry makes the system tolerant to severe distortion, and allows encoding of extra information. The method is based on region adjacency trees. After describing the mathematical foundations, we present a set of simulations to evaluate the algorithm and optimise the fiducial design.

1. Introduction

A number of vision tasks rely on the fast and robust recognition of distinctive markers or fiducials in a scene. Our interest is in augmented reality, where fiducial labels can be used to identify objects, register overlays, and calibrate camera location, but tasks in robot vision, automated assembly and virtual reality may also employ fiducials. In all cases the requirement is that fiducials are correctly recognised and located, while naturally occurring scene structure is not falsely interpreted as fiducials. Some applications are more demanding than others: there can be requirements for recognition over a range of scales, in varying lighting conditions, at arbitrary angles and with partial occlusion. At the same time, some types of fiducial are more flexible than others, affording more functionality -for example in the number of different markers that may be recognized- or more durability- for example in the kinds of distortions a marker can undergo and still be recognized.

Fiducials can be located on the basis of properties like colour, shape and pattern. Their design is bound up with the design of detection mechanisms, and so in seeking to optimize a particular approach to fiducial detection we must consider both aspects.

In this paper we consider the detection of fiducials in images by their topological properties. Our fiducials are two-level (black and white) patterns. We detect them by adaptive image thresholding, followed by region adjacency analysis. This means that neither colour nor geometry are inherent properties. In Section 2 we describe the benefits of this approach, and in Section 3 outline our algorithms, identifying the parameters that need to be optimized. We have conducted extensive simulations to evaluate different topological structure and detection algorithm parameters and these are reported in Sections 4 and 5. We briefly report a working application that uses our fiducials and detection method in Section 6.

2. Benefits of the Topological approach

Most systems proposed for real-time image fiducial tracking are based on geometrical features or colour recognition[1, 2, 3, 4, 5]. Approaches based on region adjacency graphs and graphs in general have been traditionally proposed for (off-line) batch applications, as they generally suffer from the computational complexity of the sub graph isomorphism problem[6]. This consists in finding a mapping between all the nodes of a target graph -representing the object that is searched- and a subset of a larger graph -representing the scene, i.e. the search space. The sub graph isomorphism is an NP complete problem, and hence its complexity is exponential in the size of the two graphs.

A notable exception seems to be represented by the work of Messmer and Burke[7], where subgraph isomorphism is used for Real-Time fiducial recognition.

Another problem for a topological approach arises when the fiducials are occluded. In these circumstances the occluding object's regions merge with those of the fiducial, strongly modifying its topological structure.

On the other hand, a topological approach leads to independence from fiducial geometry and hence to tolerance to deformation, as long as topology is preserved. This class of deformations covers a larger range than those allowed for algorithms based on geometrical features – such as warping. Consequently, it is possible to sketch the fiducials by hand, or to attach them to deformable objects (such as clothes).

Moreover, as the geometry of the fiducials is transparent to the recognition algorithm, it can be used to encode extra information. Fiducial alphabets or "families" can be defined so that all the symbols share the same topological structure, but have different geometries, (as illustrated in Figure 1). In this way, all the symbols from the alphabet will be recognised and located in a first pass by the recognition algorithm. Subsequently, simpler image processing routines can be applied locally to distinguish among different symbols.

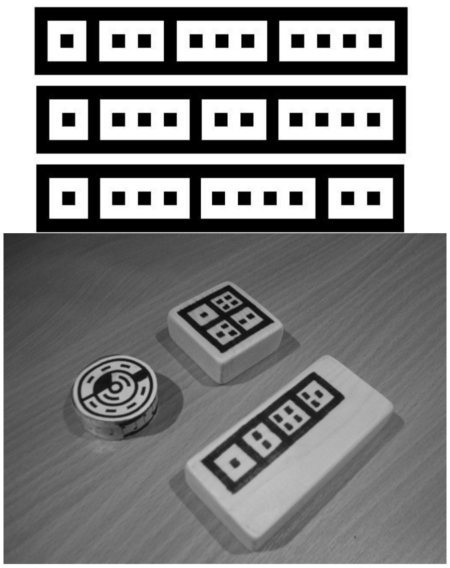

Figure 1. *(a) (top) Different fiducial markers with the same topological structure. (b) (bottom) Objects labelled with fiducial symbols*

The system that we propose benefits from the topological approach and at the same time simplifies the complexity of the search by (1) using a binary threshold to reduce the image to two levels, and (2) introducing constraints on the region adjacency structure of the fiducials. This makes our algorithm suitable for real-time applications.

Use of two level (black and white) symbols affords the maximum contrast, and therefore resilience to illumination variation across the fiducial.

A controlled amount of tolerance can also be introduced in the recognition process. This increases the robustness of the system when the topological structure of the fiducials is modified, for example because of spatial illumination changes or even moderate occlusion. To be effective, tolerance has to be defined in conjunction with the symbols' design. In general a trade-off will be required between tolerance/robustness and the amount of extra information encoded in the symbols. Details are given in the following sections.

3. Description of the Algorithms

3.1 Adaptive Thresholding

We have evaluated existing adaptive two-level thresholding algorithms due to Bernsen [7] and Niblack [8]. We report here a new approach to adaptive thresholding that outperforms both for our application.

In common with other adaptive algorithms we consider a local window on the input image and calculate a threshold for that window. Our approach is based on the idea that image peaks, ridges and the shoulders of edges should, if possible, be thresholded to white, while image pits, valleys and the feet of edges should, if possible, be thresholded to black. Within any window it will be impossible to separate all of the first type of feature from all of the second with a single luminance threshold, but we aim to find the minimum-classification-error threshold for these points. The isotropic laplacian [9] of the original image has high magnitude at points of high luminance surface curvature. These occur at peaks, along ridges, and along the shoulders of edges for high positive values of the laplacian, and at pits, along valleys, and along the feet of edges for high-magnitude negative values. We therefore threshold the laplacian at a positive level and sample the original image at all super-threshold points.

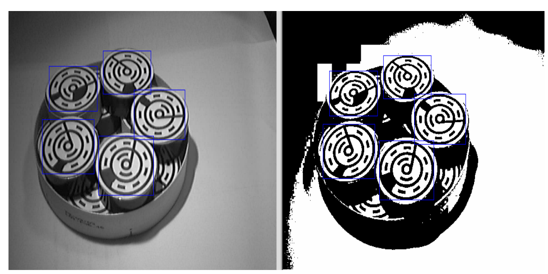

Figure 2. *Example image before and after thresholding*

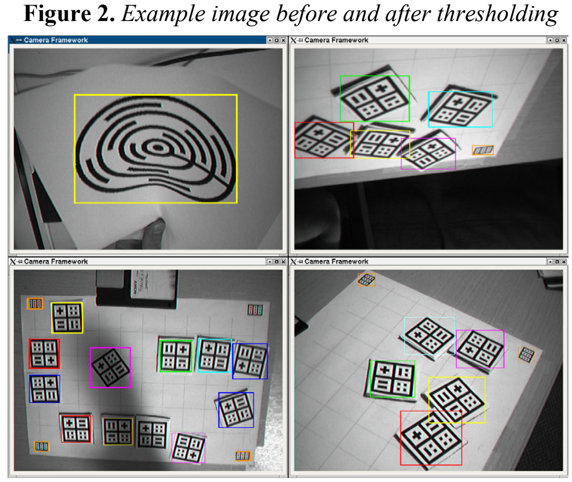

Figure 3. *Fiducials recognized by the algorithm; note warping, scale variation and recognition of different topological and geometrical structures.*

These values are taken as samples of a class that should be labelled "white". Similarly, we threshold the laplacian at a negative level and sample the original image at all sub-threshold points. These values are taken as samples of a class that should be labelled "black". The problem then becomes a 1D two class discrimination problem. We could choose the threshold at the average of the two class means, but it is computationally inexpensive to search for the minimum error (i.e. maximum a posteriori) classification threshold. This threshold is then applied to the whole region. Figure 2 shows representative results of using our approach.

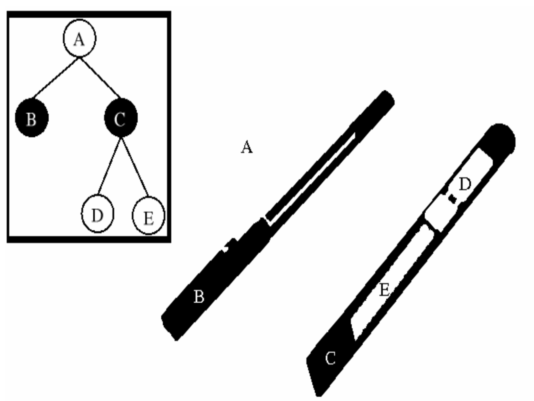

Figure 4. *Example region adjacency tree.*

3.2 Region Adjacency Tree

The core of the recognition algorithm is based on the adjacency structure of the connected regions of the image. The connected components of the image are extracted with a recursive procedure[11]. During this operation information about adjacency of regions is also collected.

The region adjacency information is stored in the form of an undirected graph G(N,A). Each of the graph nodes $n \in N$ represents a connected component of the picture. Two nodes are connected by an arc $a \in A$ if and only if they are adjacent. The number of arcs starting or ending at a node is called "degree" of that node.

It has been proven that for binary images the region adjacency graph is a bipartite tree[12]. This means that the graph is connected and acyclic (i.e. it contains no loops). Moreover the adjacency relationship for this type of picture always reduces to a "surrounds" or "is surrounded by" relationship[12], leading to an oriented tree, rooted – in principle – in the picture's "background". In practice, the "background" cannot be identified directly, so our algorithms traverse an un-rooted and non-oriented tree.

Once the region adjacency tree of the entire image is ready, the fiducial recognition becomes a sub-tree isomorphism problem. This consists in searching for a one to one correspondence θ() between the fiducial tree (nodes and arcs) and one or more subsets of the scene tree preserving adjacency, i.e. θ() must be such that $\forall\ a_k \in A$ connecting n_i and $n_j \in N$ in G, $\theta(a_k)$ will connect $\theta(n_i)$ and $\theta(n_j)$ in θ(G). This is a classical problem in computer science[10], and can be solved with a graph matching algorithm. Its complexity is then the same as the graph matching complexity: $O(n\ m^{1.5})$, where n and m are the cardinality of the scene and the fiducial trees[14].

We propose to simplify the sub tree isomorphism by imposing a number of constraints on the fiducial adjacency structure. The number of levels in the fiducial tree is limited to 3. We name these levels: root, branches and leaves, with obvious meaning. The root is connected to each of the branches and an external region. The branches are connected to the root and to a number of leaves. We also term a node of degree one connected to the root an "empty branch", because it is at the same level (and has the same colour) as the other branches. Figure 5 illustrates these terms.

With these restrictions, one fiducial with n_b branch nodes can then be represented by a sequence $\{a_k\}$ of (n_b+1) integers: one indicating the number of branches and the others indicating the number of leaves in each branch. These last n_b integers will be indicated as the set $\{l_k\}$.

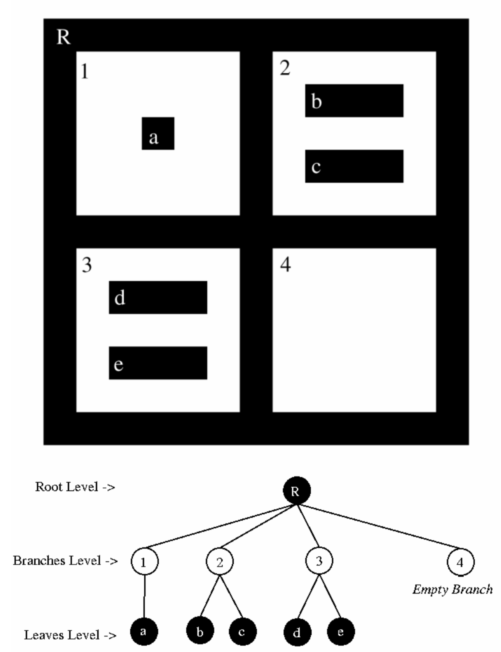

Figure 5. ***Example fiducial (0,1,2,2) and its adjacency tree.***

With this representation the recognition can be performed traversing the scene tree only once.

Every node n* of the scene tree with degree (n_b + 1) is considered a candidate root; the set {l'$_k$} of the (n_b + 1) integers will be constructed with the degree of each node connected to n*. The intersection of {l_b} and {l'$_b$} is then key to the recognition process: if the cardinality of {l_b} Ç {l'$_b$} is n_b, then n* is accepted as root of a fiducial instance.

A controlled amount of tolerance can be introduced in this recognition process. For example requiring that at least $n_b'<n_b$ branches conform to the model, or relaxing the conditions to accept a node as a candidate branch or candidate root.

In the experiments described in the following sections, we introduce tolerance on the number of branches that have to match the model. Tolerance value zero requires all the branches to be matched to the model, tolerance value 3 requires only one branch to be matched.

Rather than starting from candidate roots, the recognition can also be initiated from leaves. All the leaves of the scene tree will be labelled as such. Then, every node linked to any leaves will be labelled as branch, and similarly every node linked to any branches will be labelled as a potential root. In this way no constraint on the degree of candidate roots is introduced. Hence, the conditions for recognition are relaxed: tolerance is introduced with respect to the topology of the fiducials. With this variation the algorithm is capable of recognising partially occluded fiducials, but suffers from a drastic increase in the number of false positives. Countermeasures to limit this problem while retaining the advantages of this approach are under consideration. This second method does not allow the use of empty branches.

In this paper we will consider only fiducial trees with four branches, hence each will be identified with a sequence of four integers representing the number of leaves in each branch. The order of the sequence is irrelevant from the topological point of view. In general, however, the order of the sequence can be used to describe some geometrical property of the markers, once a convention has been established. For example the fiducials in Figure 1 (a) are identified with the sequence (1,2,3,4) (from a pure topological point of view), or can be described as (1,2,3,4), (1,3,2,4) and (1,3,4,2) if considering the order of the branches from left to right. In the rest of this article we will focus only on the topology of the fiducials, and hence the order of the leaves sequences can be ignored.

4. Experiments Description

We have designed a set of experiments to prove the validity of the system and optimise the fiducial design. Although the tests are not a simulation of the real conditions of use of the fiducial recognition system (for example the test fiducials have not been geometrically distorted), they provide a means to compare fiducial design and detection alternatives.

We have used the number of false positives (i.e. the regions erroneously classified as a fiducial symbols) and false negatives (i.e. fiducial symbols not recognized) as performance indices. For the false positives case, we have looked for a range of fiducial symbols in 10000 varied, cluttered but fiducial-free, test images. The range of symbols tested spanned all the possible combinations going from 4 to 12 total leaves, limiting the maximum number of leaves in any branch to 4. The results of this test demonstrated the various configurations' resilience to false positives.

The second experiment required synthetic symbols to be introduced in the test image, simulating realistic conditions of usage. In particular we have simulated spatial variation in the illuminations by randomly changing the contrast of part of the symbols. This consisted in either reducing the intensity of dark areas (to simulate highlight) or reducing the luminosity of light areas (to simulate shadow). For the evaluation of the results, the contrast variation has been classified in four ranges and expressed as a percentage of the maximum value (i.e. pure black and white). The four ranges are: below 25% of the maximum value , between 25 and 50%, between 50 and 75%, and above 75% of the maximum. The symbols have been partitioned randomly as illustrated in Figure 6.

Instances of the same range of symbols used in the false positives test were introduced in random positions and at random sizes in each of the test images.

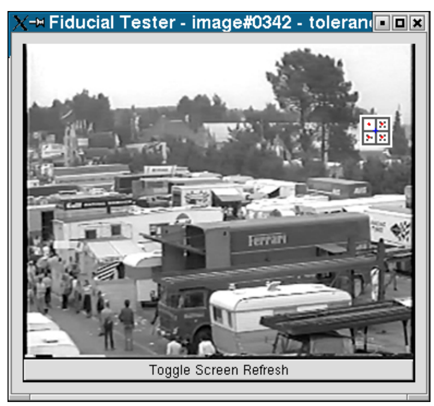

Figure 6. ***Example frame from the false negative test.***

5. Experiments Results

5.1 False Positives

Figure 7 shows the average number of false positives detected in 10,000 test images over (a) all fiducial structures (b) those that have no empty branches and (c) those that have at most one empty branch. As tolerance (defined in section 3.2) is increased, the number of false positives increases. With a tolerance of 0, there are only 14 false positives over all fiducials. A tolerance value higher than 3 causes many false positives and is clearly unusable.

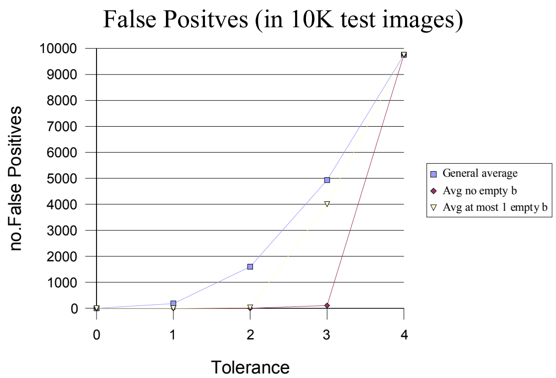

Figure 7. False positives vs tolerance.

Figure 7 shows that the presence of empty branches increases the number of false positives significantly. However, empty branches take up less space on the fiducial than branches containing leaves: Adding an empty branch to an existing fiducial always improves its performance; filling that branch with leaves improves it further, but costs significantly more in fiducial real estate. However, in the remainder of the experiments we have considered only structures containing one empty branch, at most.

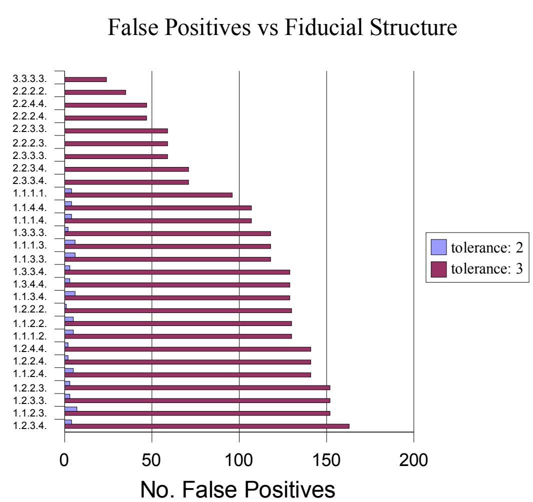

Figure 8. *False positive performance for different structures*

Figure 8 shows the false positive performance of different topological structures. Balanced fiducials -i.e. those having branches with equal numbers of leaves- perform much better than unbalanced fiducials. This leads to the recommendation that balanced fiducials should be used where possible, to avoid false detections. However, we note that unbalanced fiducials can encode information in their ordering of leaves, so may be more suitable for some applications.

5.2 False negatives

The results for the false negatives tests have been divided as follows. As explained in section 4 and illustrated in Figure 6, these tests involved inserting fiducials into images and generating an artificial shadow overlaying part of the fiducial. We found that the proportion of the fiducial that was shadowed had little effect on the results except for the cases where very little or almost all of the area was in shadow.

Figure 9 shows results for the shadow being less that 1/8 or more than 7/8 of the total area, while Figure 10 shows averages for all other degrees of shadowing. In Figure 9, as we would expect, the contrast between the shadowed and unshadowed areas does not have a significant effect on performance, whereas in Figure 10,

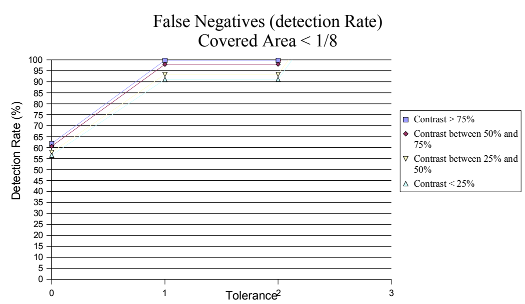

Figure 9. *False negatives for structures with no more than one empty branch, less than 1/8 of the area covered.*

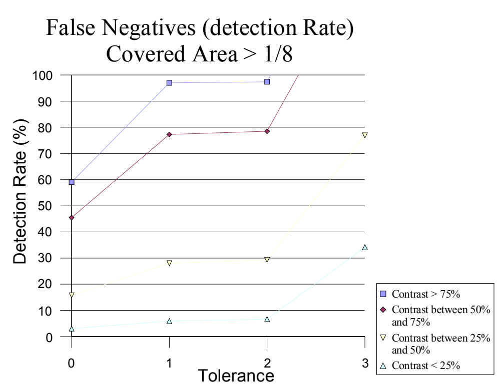

Figure 10. *False negatives for structures with no more than one empty branch, more than 1/8 of the area covered.*

it does. Figures 9 and 10 both show that false negative performance is, on average, roughly the same for tolerance 1 and tolerance 2: the higher tolerance does not yield a significantly better detection rate. Figure 9 tells us that at either of these tolerances an unshadowed fiducial, or one where the change in illumination contrast across the fiducial is limited, will be detected 99% of the time, while Figure 10 shows that 78% of fiducials are detected even when the contrast is in the range 51% - 75%. The false negative results are not immune to false positives: it can be seen that for tolerance > 2 the curves go over the total number of fiducial inserted.

We do not show false negative results for fiducial structure because there is little difference in performance over all structures.

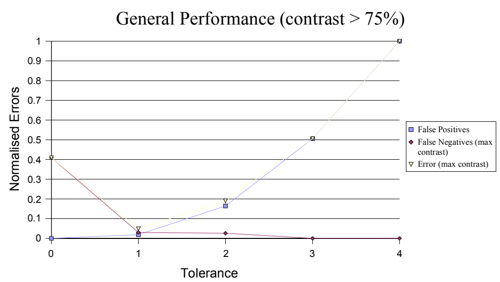

Figure 11. *Normalized false positives and false negatives, and performance index. High contrast*

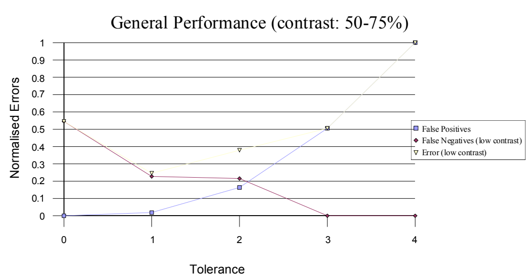

Figure 12. *Normalized false positives and false negatives, and performance index. Medium high contrast*

All tests are summarized in Figures 11 and 12. Here the normalized number of false positives and false negatives are plotted against tolerance and their sum is used as a general performance index. The minimum indicates a tolerance value of 1 as optimal. The superior performance of balanced fiducials on false positives is not cancelled by any opposite effect for false negatives, so balance is preferable, subject to the caveat about imbalance allowing coding of information through position of branches in the fiducial.

6. "d-touch": an Example Application

The system described has been employed to implement a generic framework for tangible used interfaces, as described elsewhere[14]. In d-touch, the fiducials are attached to objects to make them recognizable and localizable by a low end computer equipped with a simple web-cam. The position of the objects can then be arbitrarily mapped to any parameter. The objects lie on a surface that in this way becomes interactive. The surface can also be annotated with visual cues to relate position and meaning (or function), enforcing the idea if its interactivity.

In particular we have experimented with mapping sounds and their reproduction parameters to the object coordinates. Fiducials are also used to mark four fixed points on the interactive surface, which are detected and used as a reference frame. Then real-time detection of fiducials on interaction objects allows the performance of music (Fig 13).

On a consumer grade personal computer (Pentium 4 1.7GHz) the system works at about 8 frames per second using a resolution of 640 by 480 pels, or about 20 fps at 320 by 240 pels.

Figure 13. *The "Augmented Musical Stave" application.*

7 Conclusion

We have described a fiducial design and detection method that offers fast and robust location and tracking of markers. We have conducted systematic experiments to optimize the thresholding and topological analysis algorithms in the method. We conclude that topological fiducials provide robustness to illuminate variation and false detection. Best performance is achieved with balanced fiducials and a detection tolerance of 1. Future work will involve the development of further tests to protect against false negatives. A large set of real images containing fiducials is being collected to allow quantitative evaluation of performance in practical conditions.

References

1. Kato H. and Billinghurst M., Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System, *Proc. of the IEEE & ACM IWAR '99*, pp.85-94, 1999.
2. Cho, Y. and Neumann, U. (1998). Multi-ring Color Fiducial Systems for Scalable Fiducial Tracking Augmented Reality.InProceedings of IEEE 1998 Virtual Reality Annual International Symposium (VRAIS '98), Page 212
3. Underkoffler, J. and Ishii, H. Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface. *Proceedings of CHI '98*, April 1998.
4. Rekimoto J. and Ayatsuka Y. "CyberCode: Designing Augmented Reality Environments with Visual Tags", Proceedings of DARE 2000, 2000
5. G. A. Thomas, J. Jin, T. Niblett, C. Urquhart, "A Versatile Camera Position Measurement System for Virtual Reality TV Production", International Broadcasting Convention (IBC97), pp. 284-289 (September 1997)
6. Johnston D., Clarke A., A Vision Based Location System Using Fiducials, VVG 2003.
7. B. T. Messmer, Bunke H., "A new Algorithm for Error-Tolerant Subgraph Isomorphism Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 5, May 1998.
8. Bernsen J., Dynamic Thresholding of Grey-level Images, *Proceedings 8th International Conference on Pattern Recognition*, Paris, 1251-1255 (1986).
9. Niblack W, "An Introduction to Image Processing", Prentice-Hall, 1986, pp115-116
10. R. C. Gonzalez, R. E. Woods. "Digital Image Processing", Prentice-Hall, 2002, pp128-131
11. Reyner S. W., "An Analysis of a Good Algorithm for the Subtree Problem", SIAM Journal of computing, Vol.6, No.4 (December 1977)
12. Rosenfeld A. "Adjacency in Digital Pictures", Information and Control, No 26 pp. 24-33 (1974).
13. S. Skiena.The Algorithm Design Manual**.**Telos Pr., 1997.
14. Costanza E., Shelley S. B., Robinson. J. A. "*d-touch: a Consumer-Grade Tangible Interface Module and Musical Applications" submitted to British Conference on Human-Computer Interaction (HCI03), 2003*