Contextual Visual Localization: Cascaded Submap Classification, Optimized Saliency Detection, and Fast View Matching

Francisco Escolano, Boyan Bonev, Pablo Suau, Wendy Aguilar, Yann Frauel, Juan M. Sáez and Miguel Cazorla

***Abstract*— In this paper, we present a novel coarse-to-fine visual localization approach: *Contextual Visual Localization*. This approach relies on three elements: (i) A minimal-complexity classifier for performing fast coarse localization (submap classification); (ii) An optimized saliency detector which exploits the visual statistics of the submap; and (iii) A fast view-matching algorithm which filters initial matchings with a structural criterion. The latter algorithm yields fine localization. Our experiments show that these elements have been successfully integrated for solving the global localization problem. Context, that is, the awareness of being in a particular submap, is defined by a supervised classifier tuned for a minimal set of features. Visual context is exploited both for tuning (optimizing) the saliency detection process, and to select potential matching views in the visual database, close enough to the query view.**

I. INTRODUCTION

Once Simultaneous Localization and Mapping (SLAM) algorithms have learned maps of the environment, visual information is key for endowing autonomous robots with the ability of exploiting successfully such maps. This task implies solving other problems like: (i) Finding the position of the robot in the map (global localization)[1][2][3][4][5][6]; (ii) Tracking the position of the robot over time, for instance to supervise a given trajectory (pose maintenance, servoing)[8]; and (iii) Exploring a sequence of landmarks for returning to a given position (homing) [9]. In this paper, we focus on the global localization (robot kidnapping) problem, although some of our contributed techniques may be used for solving pose maintenance, homing, or even SLAM subproblems like loop-closing [10].

Recent methods for visual localization, closely related to object recognition approaches following the *constellation paradigm* [11][12][13], share two features. Firstly, these algorithms rely on computing a set of features invariant under scale, motion and illumination, in order to index the images (an early attempt is presented in [14]). And secondly, they tend to adopt a *coarse-to-fine* approach, in order to minimize the number of hits to the visual databases. For instance, in [1], the localization process is accelerated by building a visual vocabulary from clustering invariant features. Such vocabulary is the basis of an inverted index (accounting for occurrences of elements of the vocabulary in the image) which yields coarse localization. Finally, fine localization, among the five best candidates of coarse localization, relies on the number of matched descriptors. A subsequent verification stage exploits epipolar geometry for removing ambiguities (this is the main difference with respect to the approach presented here). In [3], which evolves from [4], the visual vocabulary is replaced by a selection of feature points in terms of their information content; localization relies on matching feature descriptors and a HMM is introduced in order to account for neighborhood relations between views. In [2], the initial matching is filtered by estimating, as in [1], the epipolar geometry through a RANSAC algorithm. RANSAC is used for global localization in [5], when 3D data is available. The problem of learning a set of features for pose estimation has been investigated in [18], and the problem of selecting the minimal set of features for navigation is tackled in [19]. Finally, a method for reducing the number of images in the data set with the minimal loss of information is proposed in [20].

Considering the latter state-of-the-art approaches to visual localization, there are few attempts of exploiting image statistics derived from filters outputs (some of them with invariant properties) in order to speed-up localization (that is, to implement coarse localization). Early attempts [6] exploit multidimensional histograms but there are few later efforts addressed to find the *minimal-complexity classifier*, that is, the classifier exploiting a minimal number of filters while yielding the minimal error. More recently [7] boosting has been exploited to build strong classifiers with range data.

In addition, when computing the fine localization through filtering an initial matching, epipolar geometry is a useful constraint but, due to the high percentage of outliers expected ($\approx 50\%$) an intense sampling effort is expected when RANSAC is applied. Although it is possible to exploit the statistics of inliers and outliers to reduce the complexity of the process, as it is done in [2], other approaches relying on *structural filtering* are useful in this fine-matching stage.

Regarding the scale-invariant detectors and features, the SIFT detector [21] is the usual choice in most of the latter works. Recent performance studies [22][23] shows that these features are well behaved in terms of distinctiveness, robustness, and detectability. Another interesting contribution derived from [22] is a discriminant classifier to select well behaved features. Another detector is the Kadir-Brady one [24], which is invariant to planar rotation, scaling, intensity

F. Escolano, B. Bonev, P. Suau, J.M. Sáez and M. Cazorla are with the Robot Vision Group, Departamento de Ciencia de la Computación e Inteligencia Artificial, Universidad de Alicante, Ap.99 E-03080, Alicante, Spain sco@dccia.ua.es

W. Aguilar and Y. Frauel are with the Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Autónoma de México (UNAM), México DF, México weam@turing.iimas.unam.mx

shift, and translation. Such detector has been used, in combination with SIFT and the MSER detector, to detect loop-closing during SLAM [27]. Affine-invariant detectors, like the Harris-affine[28], are also used in robotics [10] (for a comparison between affine methods see [26]).

Scale-invariant and affine-invariant detectors are good insofar they provide a wide-baseline stability. However, their application usually introduces a computational bottleneck in between the coarse and fine localization stages. Thus, reducing such overload is a challenging question. In this paper, we propose, and successfully test, a methodology for *increasing the performance of invariant detectors*. This methodology is interesting in the sense that such increasing of performance actually depends on the visual statistics of views associated to each submap.

We finish this section with an overview of the method (and of our contributions). Our first contribution is to design a minimal-complexity classifier (Section II) for performing coarse localization with low error. The second contribution is a method, relying on Bayesian learning, for optimizing the Kadir saliency detector by exploiting the visual statistics of each submap (Section III). Given the SIFT descriptors associated to the resulting Kadir points, we perform a fast matching free of structural noise (Section IV) which is our third contribution. Comparative results between coarse and fine localization are showed in Section V. Finally, our conclusions and future work are summarized in Section VI.

II. SUBMAP CLASSIFICATION

The 3D+2D map is derived from a long trajectory of 6DOF poses captured by a color stereo camera carried by a person traversing different sub-maps learned through Entropy Minimization SLAM [15][16][17], each one indexing a 3D point cloud and a color view. Given this huge map and a query view, such view must be properly and fast classified as belonging to one of the submaps. The total path length was $209m$, which gives a rough idea of the map scale. The path starts at our lab, follows different corridors, goes downstairs to the hall, reaches the building entrance and turns right towards a trees avenue. In this work, we have considered $N_c = 6$ connected submaps (see Fig. 1): `office`, `corridor#1`, `corridor#2`, `hall`, `entrance`, and `trees-avenue`, denoted also as C#1 to C#6 respectively. The first four are indoor (the hall is donwstairs) and the last two are outdoor.

A. Supervised Learning

For each query view I^Q to be classified, we will use a set of filters to extract the minimal number of low-level features provided that they yield the desired performance. Many of these features are invariant to illumination changes, whereas others are not so invariant but very informative.

1) Extraction of Low-level Features: The initial *filter set* is given by: (i) The Nitzberg-Harris corner detector, which is derived from the matrix $\mathbf{N}_\sigma(\mathbf{x}) = \mathbf{G}(\mathbf{x};\sigma) * \{\vec{\nabla} I(\mathbf{x};\sigma)\}\{\vec{\nabla} I(\mathbf{x};\sigma)\}^T$; (ii) Canny filter edge detector output $C(I(\mathbf{x}))$ computed from $|\vec{\nabla}_\sigma I(\mathbf{x})| = |\vec{\nabla}\mathbf{G}(\mathbf{x},\sigma) * I(\mathbf{x})|$; (iii) Gradient magnitude itself $|\vec{\nabla}_\sigma I(\mathbf{x})|$; (iv) Horizontal gradient $\nabla_{\sigma,x} I(\mathbf{x})$; (v) Vertical gradient $\nabla_{\sigma,y} I(\mathbf{x})$; (vi) Twelve hue color $\theta(\mathbf{x})$ delta filters derived from sub-sampling the hue angular and ciclic domain $[0, 2\pi]$ in twelve intervals $[\theta_i, \theta_{i+1}])$ and placing a Gaussian in their mid points, that is, $H_i(\mathbf{x}) = \mathbf{G}(\eta_i - \theta(\mathbf{x});\sigma)$, being $\eta_i = (\theta_{i+1} - \theta_i)/2$; and (vii) the stereo-based relative depth $Z(\mathbf{x}) = fT/d(\mathbf{x})$, being f the focus, and T the baseline, when disparity d is available. In the latter cases where σ is specified, a single scale was used in this work.

From the outputs of the latter filters we retain $N_f = 18$ histograms corresponding to: Cornerness N_2, which is the second eigenvector of $\mathbf{N}$, Canny-derived edge magnitude C, raw edge magnitude $|\vec{\nabla}|$, horizontal gradient $\vec{\nabla}_x$, vertical gradient $\vec{\nabla}_y$, color H_i, and depth Z. Given N_b number of bins for each histogram, the maximum number of features is $F_{max} = N_f \times N_b$. Considering both the efficiency and the performance of the subsequent feature selection process, N_b must be kept as small as possible. Furthermore, independently of the N_b, initial experiments showed that cornerness and Canny magnitude where not informative for our map, and thus, they are not considered in this paper (then $N_f = 16$).

2) SVM/K-NN Classifier: The feature selection process relies on estimating the averaged classification error for a given feature subset. As the classes (sets of views of each submap) are chosen by hand (supervised learning) we have tested both K-NN classifiers and SVMs. K-NN classification works well for $N_v = 721$ images because lazy learners (which need to keep all examples in memory) are adequate when the amount of data is not too large. In these conditions, we found that after optimal feature selection, K-NNs (with optimal neighborhood $K = 1$, experimentally found) slightly outperform SVMs 88.55 vs 86.86% of correctly classified instances, yield better a Kappa statistic (0.8602 vs 0.8393) and smaller root relative squared error (52.4% vs 84.5%). Furthermore Table I, shows that K-NNs and SVMs have yield similar classification results (in this latter tables, cells with unique values show the coincidences). However, although SVMs scale better when more complex maps are considered, in this work we will build and K-NN classifiers for two main reasons: (i) Lower-error achieved with them, and (ii) NNs are useful in order to complement the fine-localization step.

TABLE I
K-NN VS SVM CONFUSION MATRIX

	C#1	C#2	C#3	C#4	C#5	C#6
C#1	26	0	0	0	0	0
C#2	2/3	63/56	1/4	0/3	0	0
C#3	0	1/0	74/67	1/9	0	0
C#4	4/12	5/6	10/0	96/95	0/2	0
C#5	0	0	0	0	81	0
C#6	0	0	0	0	30/23	78/85

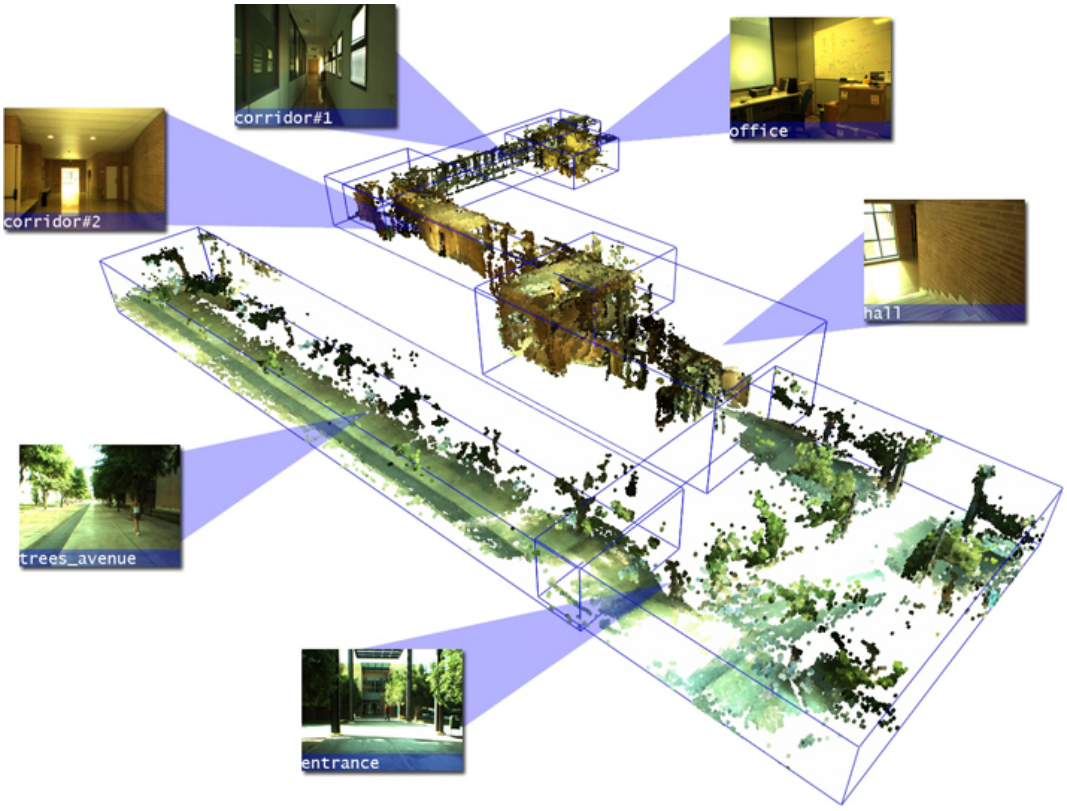

Fig. 1. 3D+2D map learned through Entropy Minimization SLAM, showing representative views of each submap.

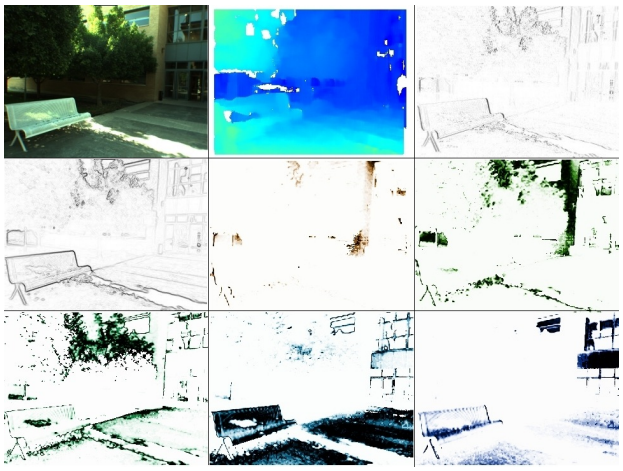

Fig. 2. Some selected filters. From Top-bottom and left-rigth: input image I, depth Z, vertical gradient $\vec{\nabla}_y$, gradient magnitude $|\vec{\nabla}|$, and the color filters: H_1 to H_5. Filters H_8 to H_{10} were also selected but not showed because they yield null output for the input image.

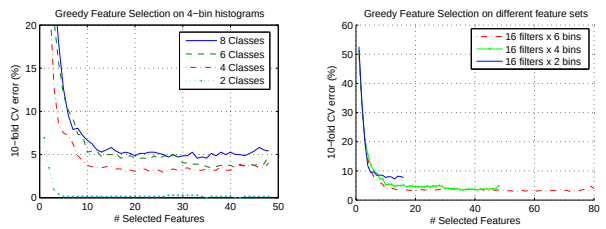

Fig. 3. Classification tuning. Left: Finding the optimal number of bins N_b. Right: Evolution of the CV error for different number of classes N_c.

B. Selection of Low-level Features

Instead of performing an exhaustive/combinatorial search, which is unpractical unless a small F_{max} is considered, we will wrap the 1-NN classifier in a greedy algorithm.

1) Greedy Wrapping: Let $\mathcal{V} = \{\mathbf{v}_1, \dots \mathbf{v}_M\}$ be the set of input feature vectors, with dimension F_{max}, associated to the training images, $\mathcal{F}$ the set of pending (to be selected) features, and let $\mathcal{S}$ the set of selected ones. Initially $|\mathcal{F}| = F_{max}$ and $|\mathcal{S}| = 0$. At each iteration of the algorithm, we pick up all $f \in \mathcal{F}$ and evaluate them. In order to do so, we first select, for each f and from the $\mathbf{v}_i$, with $i = 1, \dots, M$, the components in $\mathcal{S} \cup \{f\}$ and build a new training set $\mathcal{W}_f = \{\mathbf{w}_1, \dots, \mathbf{w}_M\}$. For each of the $|\mathcal{F}|$ training sets, each one with a different feature included, we perform $10-$fold cross validation (10-FCV) and obtain an averaged error $\bar{E}_f$ over all partition trainings and testings. The feature f^* selected in this iteration is the one which, in combination with the features yet in $\mathcal{S}$, minimizes that error. Then, f^* is removed from $\mathcal{F}$, and included in $\mathcal{S}$, and a new iteration begins. After F_{max} iterations, the feature set $\mathcal{F}$ gets empty and we register the minimal cross-validation error $\bar{E}_{min}$.

2) Selection Experiments: In order to evaluate the latter algorithm, firstly we have studied the relation between the 10-FCV error and the number of classes N_c. A high N_c is desirable in order to minimize the number of database hits needed for fine localization. In Fig. 3(left) we show, for a fixed $N_b = 4$, that the error curve for $N_c = 8$ diverges from the one for $N_c = 6$ when more than 30 features are

selected, whereas it converges to the $N_c = 4$ error curve in these situations. This indicates that a good trade-off between efficiency and classification error is to set $N_c = 6$ which is consistent with our perceptual partition showed in Fig. 1.

On the other hand, Fig. 3(right) shows that the optimal N_b for $N_c = 6$ classes is $N_b = 4$ which is consistent with early experiments [29] showing also that this optimality is more and more consistent when the number of classes increases and thus the performance of the classifier decays. Consequently, in this work we set $F_{max} = 68$, whereas the minimal number of found features was $F_{min} = 17$. Furthermore, the impact of not using Z (for instance in low-cost devices) is a reduction of $\approx 4\%$ of the classification performance. Thus, we will use 3D information in the coarse localization.

III. OPTIMIZED SALIENCY DETECTION

As we have seen, one of the benefits of submap classification is to provide a coarse localization which allows to speed-up the subsequent fine localization. As in this work such fine localization relies on a fast structural matching between the salient features of both the query I^Q and stored I_i^S images, it should be desirable to speed-up, as much as possible, the saliency-detection process. Considering the Kadir detector, we exploit the statistics from each submap to predict, with high probability, what pixels should not be explored during the scale-space analysis. Consequently, such analysis may be focused on promising pixels.

A. Optimized Kadir Detector

The optimized Kadir detector relies on finding, for each environment, a threshold $\gamma \in [0, 1]$ for discarding pixels with not-enough relative entropy to the one at σ_{max}, the maximal scale.

1) Entropy Analysis through Scale Space: The Kadir detector assumes that visual saliency may be measured by the evolution of local complexity (entropy) along scales σ or radii of pixels in the neighborhood (isotropic case). More precisely, salient points $\mathbf{x}$ have associated a peak of entropy $H(\mathbf{x}, \sigma)$ along the scale-space, and a non-zero weight $W(\mathbf{x}; \sigma)$ depending on the divergence between the respective intensity distributions (histograms) at scales σ and $\sigma - 1$. Our analysis of H reveals that entropy changes smoothly along the scale space, despite the existence of local maxima. In addition, our experiments considering 240 randomly selected images of the Visual Geometry Group database [1] (we created a test set of $240,000$ points, $10,000$ per image) show that $\Theta(\mathbf{x}) = H(\mathbf{x}; \sigma_{\max})/H_{max}$, being $H_{max} = \max_{\mathbf{x}}\{H(\mathbf{x}, \sigma_{max})\}$, helps to determine whether pixel $\mathbf{x}$ will belong to the set of salient ones or not. The higher the latter ratio (*entropy ratio*) the more salient the pixel will be along the scale space. Filtering pixels with not high enough entropy is consistent with the idea of discarding almost homogeneous regions at σ_{max}, but finding a proper threshold γ may be an image-dependent task, unless the statistics of the views of each submap are exploited, and this may be done through Bayesian learning.

2) Bayesian Optimization of the Kadir Detector: Let $P_{on}(\Theta)$, and $P_{off}(\Theta)$ be respectively the distributions (learned offline) associated to the probability of being *on* and *off* the set of salient points, defined over all ratios $\Theta \in [0, 1]$ with respect to H_{max}. Following the same methodology used in statistical edge detection [31], here we exploit the Chernoff Information [30]

$$C(P_{on}, P_{off}) = -\min_{0 \leq \lambda \leq 1}\{\log(\sum_{j=1}^{J} P_{on}^{\lambda}(y_j) P_{off}^{1-\lambda}(y_j))\}$$

, where the y_j represent the histogram bins and J their number. Chernoff Information (CI) measures how discriminable are both distributions, that is, how hard is to find an adequate threshold γ. For a given γ, we will discard $\mathbf{x}$ for scale-space analysis when $\log \frac{P_{on}(\Theta)}{P_{off}(\Theta)} < \gamma$. The error rate for the latter test decays exponentially: $\exp\{-C(P_{on}, P_{off})\}$. Furthermore, the range of valid values for a given γ is $-D(P_{off}||P_{on}) < \gamma < D(P_{on}||P_{off})$, being for instance $D(P_{on}||P_{off}) = \sum_{j=1}^{J} P_{on}(y_j) \log \frac{P_{on}(y_j)}{P_{off}(y_j)}$ the Kullback-Leibler divergence. *Any value in the latter interval is a valid threshold*, but selecting a γ value close to the lower bound results in a convervative filter which yields a good trade-off between low-error rate and high efficiency (more pruning). Efficiency may increase by increasing also γ, but error rate may also increase depending on CI, and small CI implies narrow intervals for γ.

B. Saliency Experiments

The latter considerations apply when trying to learn P_{on}, P_{off} for the complete map, which results in a too low CI (0.3201). This result suggested us to learn a different pair of distributions for each submap. Early experiments with the 12 categories of the Visual Geometry Group database yielded CIs from 0.1446 (`camel`) to 0.4728 (`airplanes`), percentages of filtered pixels from 13.31% (`camel`) to 35.98% (`cars`) depending on the γ threshold fixed. In the latter cases, the associated percentages of saved processing time range from 7.33% to 21.08%. Consequently, we exploited visual context to optimize the saliency detector for the visual localization problem. In Table II, we show: the CIs for each submap, the conservative $\gamma \approx -D(P_{off}||P_{on})$, in order to keep the disparities with respect the Kadir detector in the range of 0.2 to 4 incorrect features on average, the higher bound $D_{on-off} = D(P_{on}||P_{off})$, and the averaged percentage of filtered points in each category.

IV. FAST VIEW MATCHING

The last step of the coarse-to-fine process presented in this paper is the matching between the query image I^Q and stored ones I_i^S in order to retrieve the most probable pose of the observer in the map. In this regard, we embed the comparison of SIFT descriptors associated to the salient points in a matching process which seeks for structural compatibility by iteratively discarding *structural outliers* and finding a

[1] http://www.robots.ox.ac.uk/ vgg/

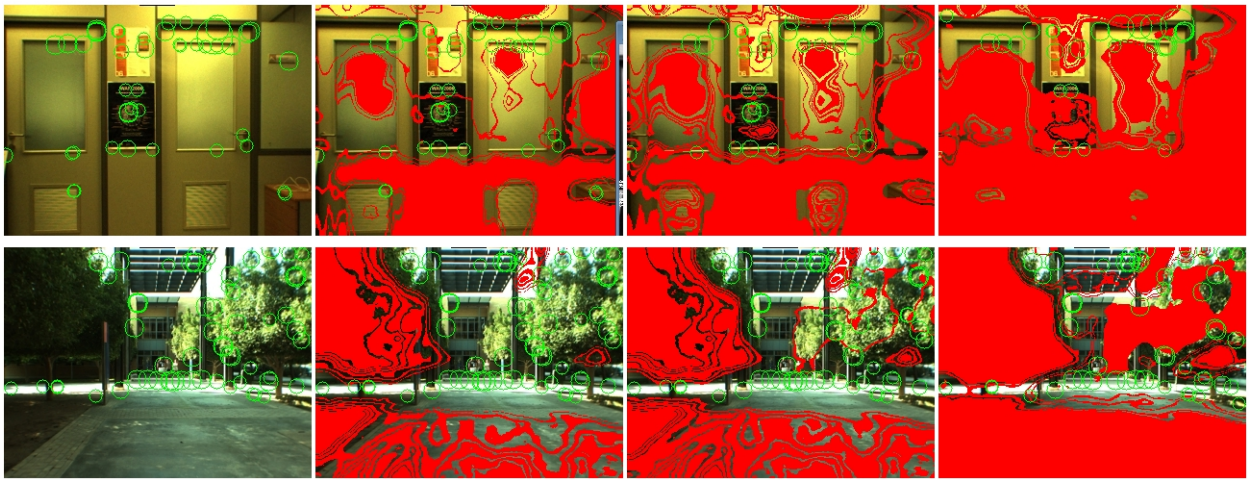

Fig. 4. Examples of pixel filtering (in red) for increasing values of γ. In both cases, the second column is the value selected in the localization experiments.

TABLE II
OPTIMIZED SALIENCY DETECTION

Environment	CI	γ	D_{on-off}	$\%Points$
office	0.8977	-9.4877	2.8305	38.51%
corridor#1	0.2482	-2.8053	1.3356	44.57%
corridor#2	0.6518	-7.4953	1.9878	60.22%
hall	0.5694	-7.4915	1.5468	44.07%
entrance	0.2859	-3.9325	0.9072	26.61%
trees-avenue	0.8543	-8.6893	3.4891	44.47%

consensus graph provided that such subgraph exists. 3D information is used only as a feature in coarse localization but *not in fine localization*.

A. One-to-one Image Matching

Given I^Q and I^S, let $\mathcal{L}_Q = \{\mathbf{s}_i\}$ and $\mathcal{L}_S = \{\mathbf{s}_j\}$ be their respective sets of salient points. Firstly, we consider their SIFT descriptors $\mathbf{D}$ and for each $\mathbf{s}_i$ we match it with $\mathbf{s}_j$ when $\mathbf{D}_{ij} = \arg\min_{\mathbf{s}_j \in \mathcal{L}_S}\{||\mathbf{D}_i - \mathbf{D}_j||\}$, and $\frac{\mathbf{D}_{ij}}{\mathbf{D}_{ij^{(2)}}} \leq \tau$, being $\mathbf{D}_{ij^{(2)}}$ the Euclidean distance to $\mathbf{s}_{j^{(2)}}$the second best match for $\mathbf{s}_i$, and $\tau \in [0, 1]$ a distinctivity threshold usually set as $\tau = 0.8$. Consequently, we obtain a set of N matchings $\mathcal{M} = \{(i, j)\}$, and we denote by $\hat{\mathcal{L}}_Q$ and $\hat{\mathcal{L}}_S$ the sets resulting from filtering, in the original ones, features without a matching in the $\mathcal{M}$ set.

B. Transformational Graph Matching

Given I^Q, let $\mathbf{G}_Q = (\mathbf{V}_Q, \mathbf{E}_Q)$ be its *median K-NN graph* computed as follows. The vertices $\mathbf{V}_Q = \{\mathbf{s}_1, \ldots, \mathbf{s}_N\}$ are given by the positions of the N salient pixels $\mathbf{s}_i \in \hat{\mathcal{L}}_Q$. A non-directed edge (i, k) exists when $\mathbf{s}_k \in \hat{\mathcal{L}}_Q$ is one of the $K = 4$ closest neighbors of $\mathbf{s}_i$ and also $||\mathbf{s}_i - \mathbf{s}_k|| \leq \eta$, being $\eta = \beta \times med_{(l,m)\in \mathbf{V}_Q \times \mathbf{V}_Q}\{||\mathbf{s}_l - \mathbf{s}_m||\}$ proportional to the median of all distances between pairs of vertices in $\mathbf{V}_Q$. Such thresholding filters structural deformations due to outlying salient points (a good balanced value is $\beta = 2$ or simply $\beta = 1$).

The graph $\mathbf{G}_Q$, which is not necessarily connected, has associated an $N \times N$ adjacency matrix $\mathbf{Q}_{ik}$ where $\mathbf{Q}_{ik} = 1$ when $(i, k) \in \mathbf{E}_Q$ and $\mathbf{Q}_{ik} = 0$ otherwise. Similarly, the graph $\mathbf{G}_S = (\mathbf{V}_S, \mathbf{E}_S)$ for a stored view I^S is build on-line (graphs are never stored, only images are stored) and has an adjacency matrix $\mathbf{S}_{jl}$, also of dimension $N \times N$ because of the one-to-one initial matching $\mathcal{M}$. Transformational Graph Matching (TGM) relies on the hypothesis that outlying matchings in $\mathcal{M}$ (typically with a percentage greater that 50%) may be removed, with high probability, by iteratively applying a simple structural criterion. Thus, TGM iterates: (i) Selecting an outlying matching; (ii) Removing matched features corresponding to the outlying matching, as well as this matching itself; (iii) Recomputing both median K-NN graphs. Structural disparity is approximated by computing the *residual adjacency matrix* $\mathbf{R}_{ij} = |\mathbf{Q}_{ij} - \mathbf{S}_{ij}|$ and selecting column $j^* = \arg\max_{j=1\ldots N}\{\sum_{i=1}^{N} \mathbf{R}_{ij}\}$, that is, the one yielding the maximal number of different edges in both graphs. The selected structural outliers are the features forming the pair (i, j^*), that is, we remove $\mathbf{s}_i$ from $\hat{\mathcal{L}}_Q$, $\mathbf{s}_{j^*}$ from $\hat{\mathcal{L}}_S$, and (i, j^*) from $\mathcal{M}$. Then, after decrementing N, a new iteration begins, and new median K-NN graphs are computed from the surviving vertices. The algorithm stops when it reaches the null residual matrix: when $\mathbf{R}_{ij} =$

$0,\ \forall i,j$. Thus, the algorithmg seeks for finding a *consensus subgraph*, and returns the number of vertices of this graph. Considering that the bottleneck of the algorithm is the recomputation of the graphs, which takes $O(N^2 \log N)$ (the same as computing the median at the beginning of the algorithm) and also that the maximum number of iterations is N, the worst case complexity is $O(N^3 \log N)$. However, the recomputation of the median graphs may be avoided by using data structures related to incoming and outcoming edges. In this latter case the overall computing time is nearly constant for all the interations.

C. Matching Experiments

We have tested the matching algorithm with several example image pairs before performing the fine-localization experiments. Early experiments with matching pairs associated to indoor images showed a 0% of errors vs the 60% of errors obtained when using a standard polynomial-cost graph-matching algorithm like Softassign [32] or its *kernelized* version, developed by some of the authors of this paper [33] in order to make Softassign more robust against structural outliers. Furthermore, the computational cost of TGM is 2-to-3 orders of magnitude lower than Softassign (it is usually bounded by 10^{-2} seconds when typically 50 matchings are considered). In Fig. 5 we show two representative examples of matchings before and after applying TGM. In the following section we will give more details about the performance in fine localization.

V. GLOBAL LOCALIZATION EXPERIMENTS

A. Coarse and Fine Localization

Contextual visual localization implies: (i) Supervised learning of the minimal-complexity classifier; (ii) Optimizing the saliency detector by exploiting statistics of each image class; and (iii) Exploiting the classifier to extract from the visual database (stored views) a set of P nearest neighbors (NNs) of the query (test) image and apply the fast matching algorithm for finding which of these P views is more consistent with the query image. Consistency is measured both in terms of similarity between local features and structural compatibility.

B. The Usefulness of Fine Localization

Is our contextual approach truly effective for global localization? The answer depends on the minimal number of $P > 1$ needed for escaping from the coarse localization results given by the case $P = 1$. In Fig. 6 (left) we show the global localization results for the test trajectory with $N_t = 472$ views vs $N_v = 721$. Such test trajectory may be considered a *ground truth trajectory* in terms of 6DOF positions but not in *visual terms*: although it was taken in similar illumination conditions to the stored one, there were dynamical events (people walking) not appearing in the stored trajectory and the temporal resolution of both sequences was also different. Both trajectories start at the small office (NW in the map) and finish at the trees avenues. We have not investigated the effect of closing-the-loop in this paper, but the success of this latter task depends highly on the view matching algorithm which supports a high number of mismatches. On the other hand, The pair of views in Fig. 6(left) shows that the features do not capture the differences between images of $C\#1$ and $C\#3$. However, the second pair shows a *back jump* from $C\#6$ and $C\#5$ because these classes are difficult to discriminate.

On the other hand, when we combine the classifier yielding the $P = 20$ NNs with the optimized saliency detection and the fast matching algorithm, we find that many of the latter *jumps* are deleted. The averaged classification time per image was 200 ms including feature extraction and finding 10 NNs; the averaged time for saliency detection depends on the environment but it is in the range 1 to 2 seconds, and the matching takes also 200 ms. The complexity is still dominated by saliency detection, although a significant reduction is achieved (the non-filtering range was 4 to 8 seconds, and after optimization we filter, from 38% to 60% of pixels). A lower choice of the number of NNs, for instance $P = 5$ or $P = 10$, does not improve significantly the performance yielded by coarse localization, so, in our system, the minimal helpful P is 20 NNs.

The latter results may be better visualized in Fig. 7, where we represent the indexes of the stored images vs the indexes of the test ones (*confusion trajectories*). Peaks in the trajectories represent jumps in the matching sequences. In the coarse case, showed in Fig. 7 (left), the confusion trajectory is very peaked even within the same environment, that is, far from the transition phases (changes of submap). However, after contextual localization, the trajectory is smoothed except at transition phases. Although no information about temporal context is exploited in this work, our results are comparable to those obtained in [3], where HMMs are used for that purpose. In addition, our test is very significant considering the large number of views tested: in [3] and in [1] less than 200 views are considered. In this work we consider 472 test images and 721 stored views.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this paper, we present a novel method for visual localization. This method relies on three elements: (i) A minimal-complexity classifier for performing coarse localization; (ii) An optimized saliency detector; and (iii) A fast

Fig. 5. Matching experiments. Left: Initial and final matchings between test image #2 test image and #45 stored image. Right: Matchings between #305 test image and #513 stored image.

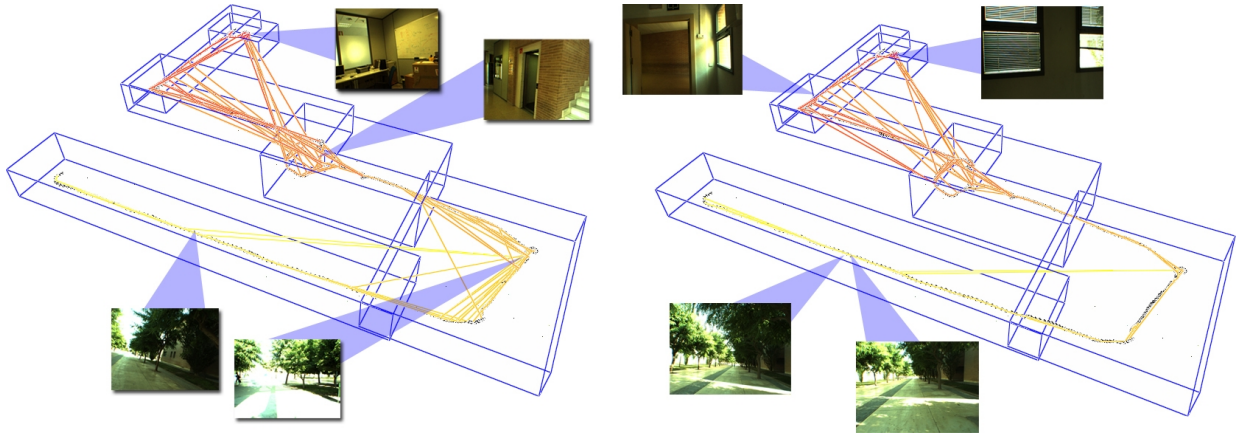

Fig. 6. Localization results. Left: Coarse localization using only the classifier. Right: Coarse-to-fine localization integrating classification retaining 20-NNs and fine fast matching. When a diagonal exists it means a confusion of 6DOF position. We show the images yielding such confusion. Sometimes they are very similar in terms of appearance.

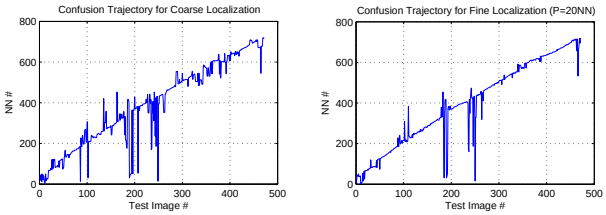

Fig. 7. Confusion trajectories for the coarse localization (left) and the integrated coarse-to-fine localization (right) after retaining 20-NNs.

view matching algorithm. These are our three contributions. Our experimental results show that the combination of these elements (contextual visual localization) is effective for solving the global localization problem with visual information. Some of the elements contributed may be exploited for solving SLAM tasks.

We have presented both representative experiments illustrating how each isolated element works, as well as global experiments showing the conditions in which the coarse-to-fine approach is truly useful. We have used a large number of views and we have not yet considered temporal context.

B. *Future Works*

This work complements our previous work in the SLAM context in the general 6DOF case but it can be extended in many ways. Our ultimate goal is to build a wearable device with mapping, localization, and navigation capabilities, in order to help blind or visually-impaired people or to be integrated in a patrolling mobile robot. Other related tasks like homing and pose maintenance are of interest. Finally, each of the contributions (minimal-complexity classifier, optimized

saliency detector and fast matching) may be improved, and temporal context will be included in a near future.

In addition, when large environments are considered K-NNs make our solution not scalable. Thus, an additional refinement, before relying on 20 NNs, is needed. For instance, we are learning indexes based on prototypical graphs (structures) for reducing the number of comparisons and improving the scalability of the method.

VII. ACKNOWLEDGMENTS

This work was supported by Project DPI2005-01280 funded by the Spanish Government, and Project GV06/134 from Generalitat Valenciana.

REFERENCES

[1] J. Wang, H. Zha, and R. Cipolla, "Coarse-to-Fine Vision-Based Localization by Indexing Scale-Invariant Features", *IEEE Transactins on Systems, Man, and Cybernetics-PartB: Cybernetics*, vol. 36, no. 2, 2006, pp. 413-422

[2] W. Zhang and J. Kosecka, "Image Based Localization in Urban Environments", *in International Symposium on 3D Data Processing, Visualization and Transmission*, Chapel Hill, NC, 2006.

[3] F. Li and J. Kosecka,"Probabilistic Recognition using Reduced Feature Set", *in IEEE Conference on Robotics and Automation*, Orlando, FL, 2006, pp. 3405-3410

[4] J. Kosecka, F. Li, and X. Yang, "Global Localization and Relative Positioning based on Scale-invariant Keypoints", *Robotics and Autonomous Systems*, vol. 52, no. 1, 2005, pp. 27-38

[5] S. Se, D. Lowe and J. Little, "Global Localization Using Distinctive Local Features", *IEEE/ESJ International Conference on Intelligent Robots and Systems*, Laussane, Switzerland, 2002

[6] T. Startner, B. Schiele, A. Pentland, "Visual Context Awareness in Wearable Computing", *in International Symposium on Wearable Computers*, Pittsburgh, PA, 1998

[7] O. Mozos, and W. Burgard, "Supervised Learning of Topological Maps using Semantic Information Extracted from Range Data", *in IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 2772–2777

[8] G. Silveira, E, Malis, and P. Rives, "Visual Servoing over Unknown, Unstructured, Large-scale Scenes", *in IEEE Conference on Robotics and Automation*, Orlando, FL, 2006, pp. 4142-4147

[9] A.A. Argyros, C. Bekris, S.C. Orphanoudakis and L.E. Kavraki, "Robot Homing by Exploiting Panoramic Vision", *Autonomous Robots* vol. 19, no. 1, 2005, pp. 7-25.

[10] P. Newman, D. Cole, and K. Ho, "Outdoor SLAM using Visual Appearance and Laser Ranging, *in IEEE Conference on Robotics and Automation*, Orlando, FL, 2006, pp. 1180-1187

[11] A. Bosch, A. Zisserman, and X. Muñoz, "Scene Classification via pLSA", *in Proceedings of the European Conference on Computer Vision*, Graz, Austria, 2006

[12] R. Fergus, P. Perona, and A. Zisserman, "A Sparse Object Category Model for Efficient Learning and Exhaustive Recognition", *IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, 2006, pp. 380-387

[13] R. Fergus, P. Perona, and A. Zisserman, "Object CLass Recognition by Unsupervised Scale-Invariant Learning", *in IEEE Conference on Computer Vision and Pattern Recognition*, Madison, WI, 2003, pp. 264-271

[14] K. Mikolajczyk and C. Schmid, "Indexing Based on Scale Invariant Interest Points", *in IEEE International COnference on Computer Vision*, Vancouver, BC, Canada, 2001, pp. 525-531

[15] J.M. Sáez and F. Escolano, "6DOF Entropy Minimization SLAM", *in IEEE Conference on Robotics and Automation*, Orlando, FL, 2006, pp. 1548-1555

[16] J.M. Sáez, A. Hogue, F. Escolano, M. Jenkin, "Underwater 3D SLAM through Entropy Minimization", *in IEEE Conference on Robotics and Automation*, Orlando, FL, 2006, pp. 3562-3567

[17] J.M. Sáez and F. Escolano, "Entropy Minimization SLAM Using Stereo", *IEEE Conference on Robotics and Automation*, Barcelona, Spain, 2005 pp. 36-43

[18] R. Sims and G. Dudek, "Learning Environmental Features for Pose Estimation", *Image and Vision Computing* no. 19, 2001, pp. 733-739

[19] P. Sala, R. Sim, A. Shokoufandeh and S. Dickinson, "Landmark Selection for Vision Based Localization", *IEEE Transactions on Robotics*, vol. 22, no. 2, 2006, pp. 334-349

[20] O. Booij, Z. Zivkovic and Ben Kröse, "Sparse Appearance Based Modeling for Robot Localization", *in IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 1510–1515

[21] D.G. Lowe, "Distinctive Image Features from Scale-invariant Keypoints", *International Journal of Computer Vision*, vol. 60, no. 2, 2004, pp. 91-110

[22] G. Carneiro and A.D. Jepson, "The Distinctiveness, Detectability, and Robustness of Local Image Features", *in IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, 2005, pp. 296-301

[23] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, 2005, pp. 1615-1630

[24] T. Kadir and M. Brady, "Saliency, Scale and Image Description", *International Journal of Computer Vision*, vol. 45, no. 2, 2001, pp. 83-105

[25] J. Matas, O. Chum, M. Urban, and T. Padjla, "Robust Wide Baseline Stereo from Maximaly Stable Extremal Regions", *in Proceedings of the British Machine Vision Conference*, 2002

[26] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, " A Comparison of Affine Region Detectors", *International Journal of Computer Vision*, vol. 65, vol. 1/2, 2005, pp. 43-72

[27] P. Newman and K. Ho, "SLAM-Loop Closing with Visual Salient Features", *in IEEE Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 644-650

[28] K. Mikolajczyk and C. Schmid, "Scale & Affine Invariant Interest Point Detectors", *in International Journal on Computer Vision*, vol. 60, no. 1, 2004, pp. 6386

[29] B. Bonev and M. Cazorla, "Towards Autonomous Adaptation in Visual Tasks", *in 7th Workshop of Physical Agents*, Las Palmas, Spain, 2006, pp. 59-66

[30] T.M. Cover, and J.A. Thomas,*Elements in Information Theory*, Wiley-Interscience, 1991.

[31] S. Konishi, A.L. Yuille, J.M. Coughlan, and S.C. Zhu, Statistical Edge Detection: Learning and Evaluating Edge Cues, *IEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no.1, 2003, pp. 57-74

[32] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching", *IEEE Transactions on Pattern Analysis and Machine Ingelligence*, vol. 18, no. 4, 1996, pp. 377-388

[33] M.A. Lozano and F. Escolano, "Protein Classification by Matching and Clustering Surface Graphs", *Pattern Recognition*, vol. 39, no. 4, 2006, pp. 539-551