Matching Feature Points in Stereo Pairs: A Comparative Study of Some Matching Strategies

Etienne Vincent and Robert Laganière

School of Information Technology and Engineering

University of Ottawa

Ottawa, ONT K1N 6N5

Abstract

Several algorithms are proposed in the literature to solve the difficult problem of feature point correspondence between image pairs. In order to obtain good quality results, they make use of different approaches and constraints to improve the quality of the matching set. A matching strategy is considered useful if it is able to filter out many of the mismatches found in an input matching set while keeping in most of the good matches present. In this paper, we present a survey of different matching strategies. We propose an empirical evaluation of their performance. The validation process used here determines the number of good matches and the proportion of good matches in a given match set and this for the different parameter values of a matching constraint.

Keywords: feature point correspondence, stereo matching, interest point detection, corners.

1 Introduction

The correspondence problem remains of central interest in the field of image analysis. Matching feature points between images constitutes a fundamental step in many computer vision applications such as the recovery of 3D scene structures, the detection of moving objects, or the synthesis of new camera views.

Although solutions for the correspondence problem are still being investigated for difficult cases, great advances has been made in establishing correspondence between views generated by an uncalibrated system of cameras. In the recent literature, the following scheme has emerged as the method of choice to perform an effective feature-based view matching [26, 31, 34]:

1. Feature points (corners) are detected [13, 22, 30].
2. Correlation is applied between feature points, then a threshold is applied to obtain a first set of candidate matches.
3. A robust method, such as RANSAC or LMedS, is used to estimate the epipolar or trifocal geometry of the camera system [9, 34].
4. Candidate pairs that are incompatible with the estimated camera geometry are rejected.
5. Guided matching using the estimated geometry is performed to find more matches.

The efficiency and accuracy of this scheme depends greatly on the quality of the candidate match set obtained in Step 2. Although usually robust to some noise, the estimator used in Step 3. requires a candidate match input set with a sufficient number of correct matches to find an accurate solution, and with a low proportion of mismatches to perform efficiently. For these reasons, another step is often added between steps 2. and 3. to filter the candidate match set. This is generally done by the introduction of some basic constraints that allow to eliminate matches that are suspected to be incorrect. These additional constraints are basic in the sense that, at the stage where they are applied, the epipolar or trifocal geometry of the camera system's geometry is not yet known. Thus image rectification, or guided matching are not an option at this point.

In this paper, we propose to empirically compare and validate the effectiveness of different matching strategies that aim to reject false matches. They will be evaluated in terms of their ability to reduce the number of false matches in a given match set, while preserving the good ones. The match set obtained in this way is then intended to serve as input to a robust estimation of the epipolar geometry which can thereafter be used to further improve the match set. It is important to note that the use of the constraints and strategies studied in this work would not be sufficient, by themselves, to obtain a match set of a quality sufficient to be suitable for most computer vision applications. Instead, they have to be used within a more elaborate matching scheme. The objective of the present study is therefore to validate the constraints used inside matching algorithms, not to study these algorithms as a whole. These algorithms have been surveyed and empirically compared in several other works [8, 11, 15, 27].

In [21], a framework to evaluate stereo vision methods is proposed. The author has also explicitly identified the constraints used in each of the studied methods. The work presented in [18] breaks down the matching process and reviews several matching methods, setting the stage for an empirical comparison, and [17] presents an algorithm for matching which exploits many constraints that are visually validated. Many authors use an iterative process in their matching algorithm before using a robust estimator. Relaxation is such an iterative process [34]. In this case, an energy function, corresponding to some aggregate value of a constraint applied to the pairs in a candidate set, is iteratively minimized. Testing the same constraint outside of such an iterative scheme represents a good way to establish its effectiveness. This why we have chosen to limit the scope of this study to the direct application of constraints.

Finally, it worths noting that many authors have presented different approaches for dealing with large changes in scale and rotations, or with the deformation induced by important change in viewpoints (i.e. with wide baselines) [3, 23, 28, 32]. To restrict the scope of this study, we chose to limit ourselves to constraints that are meant to deal with images where the change in viewpoint is limited, e.g. the case

of stereoscopic images. For the image pairs that this work intends to deal with, the methods cited above would be expected to produce inferior results as invariance to certain image transformations is necessarily obtained at the expense of some discriminating power.

The next section describes our scheme for evaluating matching methods. Then, section 3 studies the role of feature point detection in matching. Section 4 looks at the way in which correlation is applied. Section 5 is concerned with matching constraints that require corresponding features to have similar properties. Section 6 looks at matching constraints that require matches to have similar disparities as their neighbors. Finally, section 7 justifies the use of some matching constraints when the goal is fundamental matrix estimation. Section 8 is a conclusion.

2 Validating Point Correspondences

The feature point matching problem consist in finding points, in different images, that correspond to the same scene element. These matches are obtained from a large set of candidates selected in each image. In order to evaluate and compare various matching strategies, we propose to use image pairs for which all possible good matches, among a fixed set of detected feature points, have been previously identified. It will then be possible to compare the result of the application of a given matching approach against this exact solution.

For our experiments, the six image pairs shown in Figure 1 and 2 have been selected. The camera displacement in the image pairs *Kitchen* and *Lab* consists mainly of a translation (smaller in the case of *Lab*), while in *Church* it involves some rotation and in *Building* the rotation is even more significant. In *House* the disparity is mainly caused by a zoom, and finally, in *Objects* there is a significant change in the lighting conditions. Feature points have been detected on each image (see section 3). If N corners are detected in the first image and N' in the second, then $N \times N'$ possible matches are to be considered. To produce the exact set, these $N \times N'$ matches must be examined to identify all the good matches. However, because of their great number, it would be very laborious to visually examine all these possible matches.

Fortunately, to build this *ground truth* set, not all pairs need to be considered. First, it is easy, by visual inspection, to determine the horizontal and vertical disparity range of each image pair. All possible matches having a disparity outside this range can be rejected. Secondly, the matches that do not agree with the epipolar geometry of the image pair can be automatically eliminated. To this end, the epipolar geometry of each image pair was estimated using the method described in [26]. Following this pruning, we are left with a smaller set of image point pairs from which all the good matches can be extracted manually

in reasonable time[1].

Having identified the set of all possible good matches between detected points in an image pair, it becomes possible to evaluate the effectiveness of different strategies used for matching: a matching constraint is considered useful if it filters out many mismatches found in an input matching set, while preserving most good matches.

A given method will use different parameters or thresholds towards accepting or rejecting a given candidate match. There will be, usually, a tradeoff in the selection of these parameters. In order to appreciate the effectiveness of an approach, results will be shown on a graph showing the number of good matches in the resulting match set (on the Y-axis) versus the proportion of good matches in that set (on the X-axis). For each image a curve is generated representing results obtained for different values of a control parameter associated with the method under study. In such a graph, a perfect method would be one producing a horizontal line, i.e. all points eliminated being false matches, thus increasing the proportion of good matches as the filtering becomes more severe. Conversely, a useless technique would be one that produces a vertical line, i.e. one that randomly eliminates points, thus keeping the good match proportion constant. This is illustrated in Figure 3. Note that, in practice, we might expect that an effective method would produce a nearly horizontal curve until some point where the curve will start to drop vertically, when more severe thresholding cannot further improve the quality of the match set.

3 Corners as Points of Interest

The choice of a feature point detector certainly has an impact on the results produced by a given matching scheme. Several corner detectors have been proposed in the past, [30] and [22] review some of them. Among the most popular are the ones based on the *Plessey* operator [12] defined with the following matrix:

$$\begin{bmatrix} \langle (\frac{\delta I(x,y)}{\delta x})^2 \rangle & \langle \frac{\delta I(x,y)}{\delta x} \frac{\delta I(x,y)}{\delta y} \rangle \\ \langle \frac{\delta I(x,y)}{\delta x} \frac{\delta I(x,y)}{\delta y} \rangle & \langle (\frac{\delta I(x,y)}{\delta y})^2 \rangle \end{bmatrix} \quad (1)$$

where $\langle I \rangle$ denotes a smoothing operation on I. By diagonalizing this matrix, we obtain the two principal gradient directions (given by the eigenvectors) and their respective strength (given by the eigenvalues). A corner can therefore be defined as a point where the magnitude of both these directional gradients is high. Noble [25] and Harris[13] have defined operators that exploit this idea, using the determinant and the trace of matrix (1).

[1]These images, the detected feature points and the correct match set are available at www.site.uottawa.ca/research/viva/projects/imagepairs/

The Intel OpenCV library[2] proposes an implementation based on explicit eigen decomposition. In this case, a corner point is identified where the smallest eigenvalue of matrix (1) is greater than a given threshold[3]. In addition, non-maxima suppression should be performed in order to eliminate clusters of corners.

When feature points are detected for the purpose of matching, the key property of the detector is repeatability: in different views of the same scene, the detector should extract the same points, despite the variations due to a change in perspective or in the lighting conditions. The Harris detector is usually considered as the most stable detector with regards to this property [29]. To confirm this result, we compared it to the SUSAN corner detector [30], another well-known detector that uses a different strategy to identify corners. This operator is based on the computation of the area, inside a circular window, having similar brightness to the center point.

These corner detectors were used to extract approximatively 500 points in each image pair shown in Figure 1 and 2. Then, all good matches among these corners were found using the technique described in Section 2. From this, the number of scene corners that have been correctly detected in both images (the repeatability of the corner detector), was obtained. Comparing these numbers between Harris and SUSAN for an equal number of detected features results in a repeatability approximatively 3 times superior for Harris. For example, in the case of the *Kitchen* image pair, using the corners detected with the OpenCV-Harris detector, 148 correct matches were found, while SUSAN only produced 48 (cf. Figure 4). Consequently, the OpenCV-Harris detector seems indeed to be more stable.

The OpenCV library includes a second function[4] for corner detection, which filters the corners found by the first one. It ensures that the detected corners are far enough from one another. This is done by iteratively finding the strongest feature point, and throwing out all other feature points that are closer than a threshold distance from it. In order to determine if this method brings an increase in the quality of the candidate match set, the corners detected in both images of each pair were again counted, for different distance thresholds. The resulting graph, shown in Figure 5, demonstrates that this 'cleaning' of the corner set significantly worsens the set of candidate matches. Thus, this function should not be used when corners are detected for the purpose of matching between different views.

The OpenCV-Harris operator will therefore be the one used in our experiments. However, it should be noted that, when using it, the value of the threshold imposed on the eigenvalues directly influences

[2]freely available at developer.intel.com

[3]This function is called cvCornersEigenValsandVecs.

[4]This second function is called cvGoodfeatureToTrack.

the number of corners detected. To determine the effect of modifying the threshold on the quality of the candidate set, corners were detected using different thresholds on the test image pairs, and the number of corners detected in both images was determined. Results are shown in Figure 6. This graph shows that, within a reasonable range, the proportion of detected corners remain relatively constant. It therefore follows that one can increase the number of good matches just by accepting more corners. However, this is done at the price of a proportional increase of the total number of corners to analyze. The corner detector's threshold should therefore be set so that the number of matches found is suitable, but not much greater than the amount needed for the considered application.

4 Correlation

Correlation is the basic mean by which interest points of different images are matched. Correlation functions measure the similarity between point neighborhoods. Different correlation functions are presented in the following papers [1, 4, 5, 33, 34].

4.1 Average Squared Difference Correlation

A first simple correlation function consists of the average square difference (ASD) of the intensities of corresponding pixels, performed between neighborhoods of the points. ASD correlation is defined for a pair of points m and m' as:

$$ASD(m, m') = \frac{1}{N} \sum_{n \in \mathcal{N}(m), n' \in \mathcal{N}(m')} [I(n) - I'(n')]^2 \tag{2}$$

where N is the size of the neighborhood $\mathcal{N}()$.

Two basic parameters influence the performance of the correlation: the size of the window (the neighborhood) used to correlate point pairs, and the threshold value on which the decision to accept or reject a match is based. The results shown in Figure 7(a) and (b), where ASD is applied to the image pairs of Figure 1 and 2, illustrate how these parameters affect the quality of the resulting match set. As expected, tightening the threshold increases the proportion of good matches, but at the same time, decreases the number of good matches quite rapidly.

The experiment also shows that increasing the size of the window is an effective mean to identify and reject more false matches (an observation also made in [24]), but this is only true up to a certain size (until about 11×11, in our experiments). We should note that the ASD correlation was completely ineffective

on the *Objects* image pair, where there is an important change of illumination. A normalizing factor must therefore be introduced to cope with this problem.

4.2 Variance Normalized Correlation

Variance normalized correlation (VNC) is another commonly used correlation function. It offers the advantage of producing stable and reliable results over a wide range of viewing conditions. VNC is defined for a candidate match (m, m') as:

$$VNC(m,m') = \frac{1}{N\sqrt{\sigma_I^2(m)\sigma_{I'}^2(m')}} \sum_{n\in\mathcal{N}(m),n'\in\mathcal{N}(m')} [I(n) - \overline{I(m)}][I'(n') - \overline{I'(m')}] \quad (3)$$

where $\overline{I(m)}$ and $\sigma_I^2(m)$ are respectively the mean and the variance of the pixel intensities over the neighborhood of point m. The fact that VNC scores are normalized to a range $[-1, 1]$, is an advantage over other correlation functions, as it makes the choice of a threshold much easier.

Results of the application of VNC, shown in Figure 7(c) and (d), demonstrate a behavior similar to the one of ASD, except for the *Objects* pair where VNC did not fail when faced with a change of illumination, as did the ASD correlation. For any choice of parameters, the results obtained by VNC are always superior to the ones of ASD. For example, by comparing Figure 7(a) and (c), and excluding the *Lab* and *Objects* images, we see that the average rate of reduction of the number of good matches in the case of ASD is about 22.6 (meaning that when the proportion of good matches is increased by 10%, the absolute number of good matches is reduced by 22.6). For VNC, this rate is approximatively 14.4 on average. Our experiments here simply confirms a well known fact that VNC, although more complex, is superior to ASD. Considering the computational power now available, VNC should always be the method of choice when correlating windows.

4.3 Unicity

So far, all pairs having a correlation score above some threshold value were considered. Thus, a feature point could be matched with several others. Imposing unicity means that for each feature point in one image, only its strongest match in the other image is considered. A generalization of unicity was studied, where the n strongest matches are kept.

Figure 8(a) shows the results of applying VNC to our image pairs while imposing unicity of different orders. Unicity proved beneficial, as it rejected many mismatches. This important improvement obtained in the proportion of good matches is at the expense of a fairly small loss in the absolute number of good

matches. The resulting large improvement in the quality of the match sets should justify the general use of unicity of order 2 or 1.

4.4 Symmetry

When unicity is imposed, VNC becomes asymmetric, a situation which is physically impossible. Thus, a right image point, which gives the highest correlation score, when paired with a certain left image point, can itself be paired with a different left image point with a higher score.

Imposing symmetry means keeping only pairs in which each point is the other's strongest match [10]. This increases the chances that the two points in the matched pairs correspond to projections of the same physical scene point. Figure 8(b) shows the results of the same experiment as in 8(a), but where the symmetry constraint has been applied in addition to unicity. It shows that imposing symmetry is clearly advantageous as it eliminates many mismatches while affecting only few good ones. Note that symmetry is generally imposed after unicity even if, in fact, symmetry can hold for matches that violates a unicity of a given order. This is often the case for scenes where several occlusion boundaries can be found; the *church* image pair is such an example where more than 20 good matches are lost when symmetry is applied after unicity of order 1 rather than order 8.

Nevertheless, imposing unicity and symmetry constitutes a very effective way of improving the quality of a match set. The fact that these constraints have a relatively low computational costs reinforces this statement.

For this reason, the experiments presented in the remainder of this work will use match sets obtained from VNC on 9×9 windows and using a threshold of 0.8 with first order unicity and symmetry applied. Table 1 summarizes the characteristics of the resulting match sets. The question now is to determine how the quality of these match sets can be further improved.

5 Using Feature Point Properties

Many properties can be used to describe image points. Some should remain relatively unchanged in different views, and hence could be used to constrain matching.

Image pair	Proportion of good matches	Number of good matches
Kitchen	54.5%	78
Building	48.7%	56
Church	56.2%	63
Lab	67.8%	192
House	50.7%	37
Objects	67.5%	81

Table 1: Characteristics of the match sets obtained from VNC correlation using a 9×9 window and a threshold of 0.8 on which unicity of order 1 and symmetry have been applied.

5.1 Using corner characteristics

Curvature, gradient magnitude and direction, and orientation of incident edges are generally considered as good invariant feature characteristics. In [7], the gradient direction and the curvature are used among other constraints in a correlation scheme. Other such constraints are described in [2].

The principal curvatures of a point are given by the eigenvalues of matrix (1), λ_1 and λ_2. In [19], they are used to define the *cornerness* at a point P as

$$c_P = \|\lambda_1^2 + \lambda_2^2\| \tag{4}$$

The ratio of the cornerness of 2 corners is then used as a correlation function.

A corner orientation constraint can also be defined using matrix (1). Its eigenvectors correspond to the two main gradient directions, and thus, represent the two edges of the detected corner. The corner orientation can then be defined as the bisector of these two directions. This orientation can be used to constrain matching by requiring that corresponding corners have a similar direction.

Figure 9 shows the results of applying the constraint on cornerness, (a), and the one on corner orientation (b). It is seen that these constraints do not improve the candidate match set. This is not surprising since such properties are indirectly taken into account by correlation. Thus, such properties should not be used in conjunction with a correlation function that enforces similarity between the corner neighborhoods.

5.2 Corner Shape Similarity

Another possible strategy is to require that the corners in a pair have similar shapes. A corner shape is defined here as a small area around the feature point, belonging to the same scene object as this feature point. A method to extract the corner from its background is therefore required. Two such simple methods were investigated.

The first method uses univalue segment assimilating nuclei (USANs), as described in [30]. The idea is to extract the portion of feature point neighborhoods that is of similar intensity values. A USAN is then assumed to belong to the same scene object as the feature point.

The other method is inspired from rudimentary block truncation coding [6]. The correlation window is separated into two regions according to the window's average intensity value. The foreground consists of the pixels within the same region as the feature point.

Once corner shapes have been extracted, the Hamming distance between the obtained binary foreground/background maps is computed and pairs for which this distance is above some threshold are eliminated. Results are shown in Figure 10(a) and (b). While foreground extraction using USAN does not seem to be very effective, results based on truncated blocks show a certain improvement in the proportion of good matches, but the corresponding reduction in the number of good matches might appear excessive for some applications.

5.3 Eliminating the Background

Some correlation functions, such as [5], attempt to consider only the scene objects on which the feature points lie, to establish correspondence. The foreground extraction methods of subsection 5.2 can be used to determine the region to which correlation should be restricted.

Results obtained when using this kind of selective correlation (where VNC is applied directly to the foregrounds of corners, but is weighted by a multiplying factor when applied to the background) are shown in Figure 11(a) and (b). Clearly, these graphs show that performing a correlation on the foregrounds gives better results than using the simple shape similarity criterium of the previous subsection. Higher proportions of good matches can be achieved with higher multiplying factors, but at the cost of reducing their number.

6 Enforcing Disparity Consistency

It is reasonable to assume that, in most cases, the disparity of a match should be similar to the ones of its neighbors. Hence, constraints could be established that ensure that matches behave as their neighbors.

6.1 Confidence Measure

Based on the principle that each point of a match pair should have a neighborhood with similar properties, a confidence measure was proposed in [34]. It is defined for a pair of points (m, m'), using the feature points belonging to their neighborhoods $N(m)$ and $N(m')$:

$$CM_{mm'} = c_{mm'} \sum_{n \in N(m)} \max_{n' \in N(m')} \frac{2c_{nn'}\delta(m, m'; n, n')}{2 + d(m, n) + d(m', n')} \tag{5}$$

where $c_{..}$ is the correlation score (e.g. VNC), $d(.,.)$ is the Euclidean distance, and

$$\delta(m, m'; n, n') = \begin{cases} e^{-r/\varepsilon_r} & \text{if}(n, n') \text{ is a candidate match and } r < \varepsilon_r \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where r is a relative distance difference

$$r = \frac{2\,|d(m, n) - d(m', n')|}{d(m, n) + d(m', n')} \tag{7}$$

Candidate matches, found in the neighborhood, having a relative position similar to the pair being considered, are counted by the measure.

Figure 12 shows the results of constraining the confidence measure with $\varepsilon_r = 10$ and 61×61 neighborhoods. A drawback of this measure is that it cannot be estimated if a point does not have close neighbors in the candidate set. Also, the method was found difficult to tune because several parameters have to be adjusted. Although this constraint achieves positive results on some of the image pairs, it is seen that it does not perform as well as the other constraints of this section.

6.2 Disparity Gradient

The disparity gradient is a measure of the compatibility of two pairs [20]. If two pairs (m, m') and (n, n'), have disparities $d(m, m')$ and $d(n, n')$ respectively, then their cyclopean separation, $d_{cs}(m, m'; n, n')$, can be defined as the vector joining the midpoints of the line segments $\overline{mm'}$ and $\overline{nn'}$, and, their disparity gradient is defined as:

$$\Delta d(m, m'; n, n') = \frac{|d(m, m') - d(n, n')|}{|d_{cs}(m, m'; n, n')|} \tag{8}$$

Compatibility measures, such as the disparity gradient can be used in a iterative process, as in [26], where incompatible matches are iteratively removed until all pairs have a similar disparity gradient. Here, the disparity gradient as well as the constraints of the next subsection are used in a new way, in a local constraint that enforces that a match's disparity be similar to those of it's closest neighbors.

This measure was used in a constraint that accepts pairs that share a disparity gradients below some threshold value with at least 2 of its 5 closest neighbors. Figure 13 (a) shows the results of applying this constraint with varying threshold, and demonstrates that it can eliminate a significant number of outliers while eliminating few good matches.

A second experiment was performed to evaluate the effect of a change in the proportion of the neighbors that must be compatible with a match, in order for it to be considered valid. Figure 13 (b) shows the result of applying the constraint with a threshold of 0.4, but where 2 out of n neighbors must be compatible, for different ns. This shows that this constraint is most effective when applied to the 3 to 5 closest neighbors.

6.3 Relative Positions of the Neighbors

In a similar way to what may be done using the disparity gradient, the relative position of two pairs can be constrained. For good matches, the vectors $\overrightarrow{mm'}$ and $\overrightarrow{nn'}$ should be similar. Thus, a constraint was used, that requires the angle between the two vectors to be smaller then a given threshold, for at least 2 of a pair's 5 closest neighbors. Another constraint, that requires the ratios of the vector magnitudes for 2 of the neighbors to be within certain bounds, was also tested.

Figure 13 (c) shows results of applying the constraint on angles, and (d) shows the constraint on magnitudes. As long as the image pair's baseline is relatively small, these simple constraints give similar results as the constraint on the disparity gradient. However, it is more difficult to select a good thresholds on the disparity angle and magnitude.

7 Computing Fundamental Matrices from Match Sets

The matching strategies presented in this work aim at improving match sets by filtering out bad matches. The legitimacy of this objective will now be demonstrated by showing how a good initial match set can greatly improve the efficiency of fundamental matrix estimation (step 3, mentioned in Section 1).

The fundamental matrix is usually found using a RANSAC scheme, in which random selections of 8 matches are iteratively considered [14]. A fundamental matrix is computed for each selection, and its

accuracy is assessed by considering the cardinality of the subset of the candidate matches that support it. After a sufficient number of random selection, it is expected that an accurate estimate of the fundamental matrix will be uncovered. The number of iterations required is basically a function of the proportion, p, of good matches in the considered set. The number of iterations, n, needed to obtain a correct fundamental matrix with 95% probability is expressed theoretically as

$$n = \frac{\log(0.05)}{\log(1 - p^8)} \tag{9}$$

Table 7 presents the characteristics of different match sets which could be used for fundamental matrix estimation, for the 6 image pairs of Figure 1 and 2. The first two lines of each row correspond to the sets obtain using only VNC, with thresholds of 0.8 and 0.9 respectively. The last line corresponds to the match set obtained with the 0.8 VNC threshold, on which the additional constraints of unicity and symmetry were imposed, as well as a background elimination constraint using truncated blocks with a threshold of 0.9 and a background multiplier of 0.25, and a disparity gradient constraint with a threshold of 0.4.

It is seen that simply using VNC with a low threshold can yield poor results, and that the most advantageous way of improving the match set is to filter it, rather then simply increasing the VNC threshold.

We also used the Roth's software[5], described in [26], to estimate fundamental matrices from the match sets of table 7. Figure 14 shows some results which illustrate how the use of matching constraints makes the process of robust fundamental matrix estimation more efficient. The solution found in Figure 14(d), where the match set on which several constraints have been applied was used, is indeed close to the correct solution shown in Figure 14(b). This solution was found in less than 500 RANSAC iterations. This is in contrast with the solutions found using a non-filtered match set that required, in this experiment, about 10000 iterations before finding an acceptable, but still inferior solution.

8 Conclusion

It can be seen, from the experiments described here, that many methods are useful in improving the quality of sets of candidate matches. To summarize, the Harris corner detector was confirmed as a stable feature point extractor. It was then shown that VNC produces a good first candidate match set, when unicity and symmetry are imposed. Also, constraints that use simple models of the corners were investigated and

[5]available at www2.vit.iit.nrc.ca/~gerhard

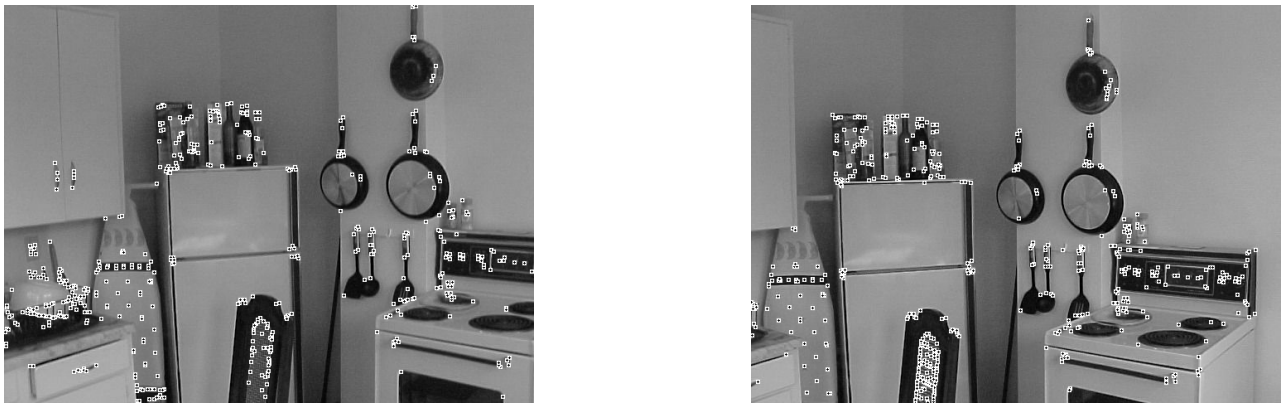
Match set	**Proportion of good matches**	**Number of good matches**	**Expected number of iterations**
Kitchen, VNC 0.8	20.0%	96	1 170 207
Kitchen, VNC 0.9	45.7%	59	1 564
Kitchen, VNC 0.8 + constraints	59.8%	70	181
Building, VNC 0.8	16.7%	83	4 951 418
Building, VNC 0.9	36.1%	48	10 407
Building, VNC 0.8 + constraints	65.3%	47	90
Church, VNC 0.8	12.3%	112	55 414 078
Church, VNC 0.9	19.1%	42	1 697 813
Church, VNC 0.8 + constraints	76.8%	43	24
Lab, VNC 0.8	27.4%	201	93 705
Lab, VNC 0.9	53.7%	195	431
Lab, VNC 0.8 + constraints	71.7%	195	42
House, VNC 0.8	13.2%	72	33 246 450
House, VNC 0.9	22.6%	36	433 770
House, VNC 0.8 + constraints	63.9%	23	107
Objects, VNC 0.8	29.8%	87	48 240
Objects, VNC 0.9	90.3%	56	6
Objects, VNC 0.8 + constraints	80.0%	80	17

Table 2: Characteristics of different match sets and the theoretical expectation of the number of iterations required to find the exact fundamental matrix using RANSAC.

found to be beneficial. Finally, comparing matches with their neighbors to ensure that they have similar disparities was shown to allow the rejection of many mismatches.

A disadvantage of combining many constraints is that it results in more thresholds, and thus a greater need for tuning. However, as seen in the example of Table 7, good, though suboptimal, results can be obtained by combining constraints using conservative thresholds.

In the last section, it was demonstrated that the use of effective matching constraints can greatly improve the process of fundamental matrix estimation. This work was restricted to the study of matching in the case of image pairs with relatively small baselines, an interesting extension would be to investigate the strategies that work best in the case of image pairs that are more difficult to match.



(a) Kitchen

(b) Building

(c) Church

Figure 1: Three of the six test image pairs with feature points extracted (approximatively 500 per image).

(a) Lab

(b) House

(c) Objects

Figure 2: Three of the six test image pairs with feature points extracted (approximatively 500 per image). The *Lab* image pair is from University of Tsukuba.

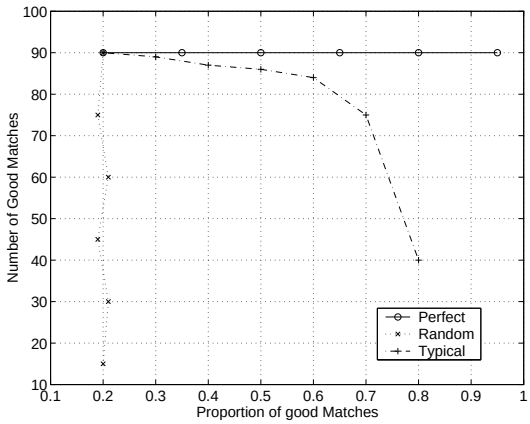


Figure 3: The evolution of the number of good matches, in a given match set, versus their proportion in that set. The three curves correspond to cases of methods behaving perfectly, randomly, or typically, as discussed in the text. The common starting point corresponds to an initial set containing, in this fictitious illustrative example, 90 good matches representing 20% of the total set. Data points on each curves correspond to different parameter selections, filtering being more severe as we move bottom-right.

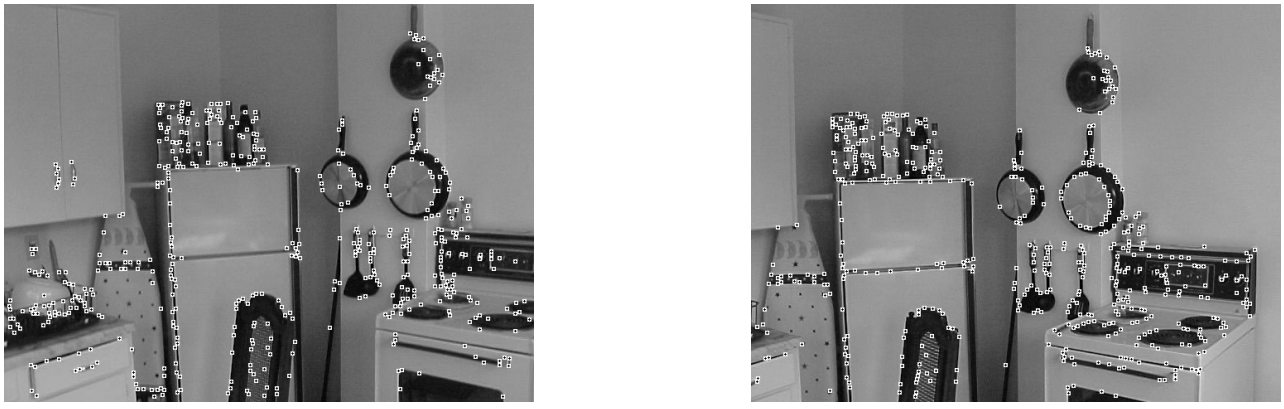


Figure 4: The corners extracted by the SUSAN corner detector, for the *Kitchen* image pair (approximatively 500 per image).

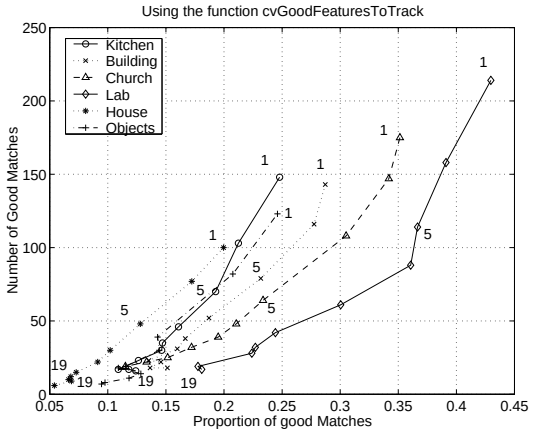


Figure 5: Eliminating corners close to stronger ones. The numbers shown represent the minimal acceptable distance between corners.

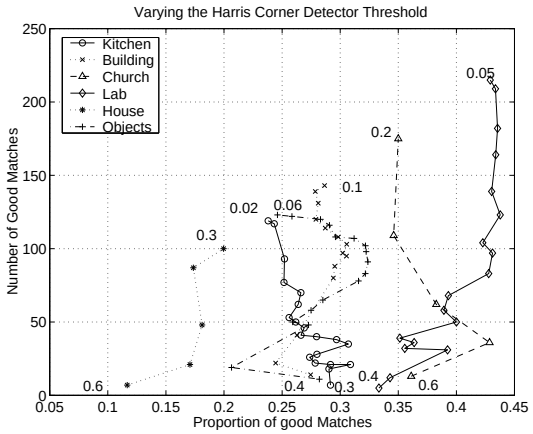


Figure 6: Modifying the threshold value of the Harris corner detector.

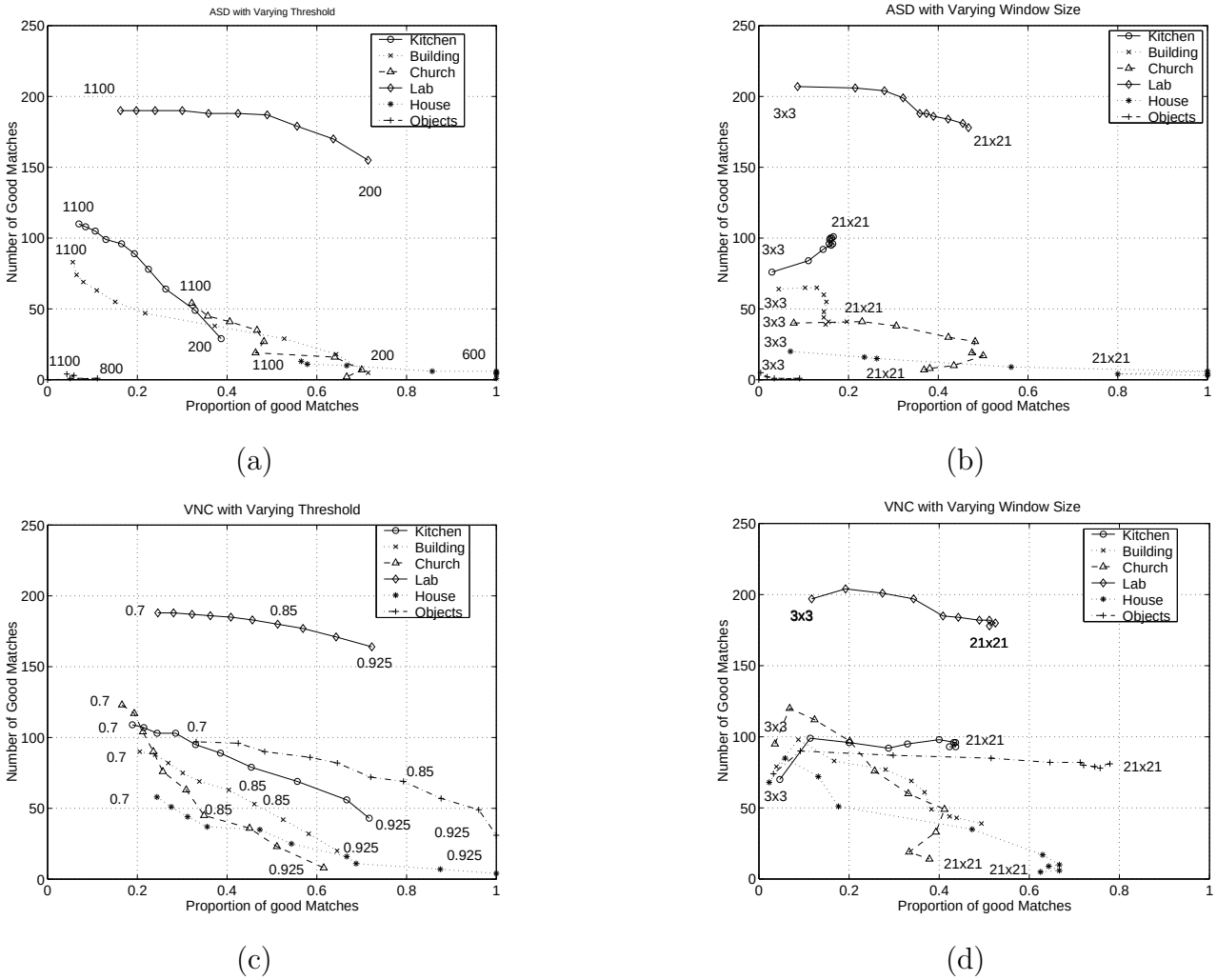
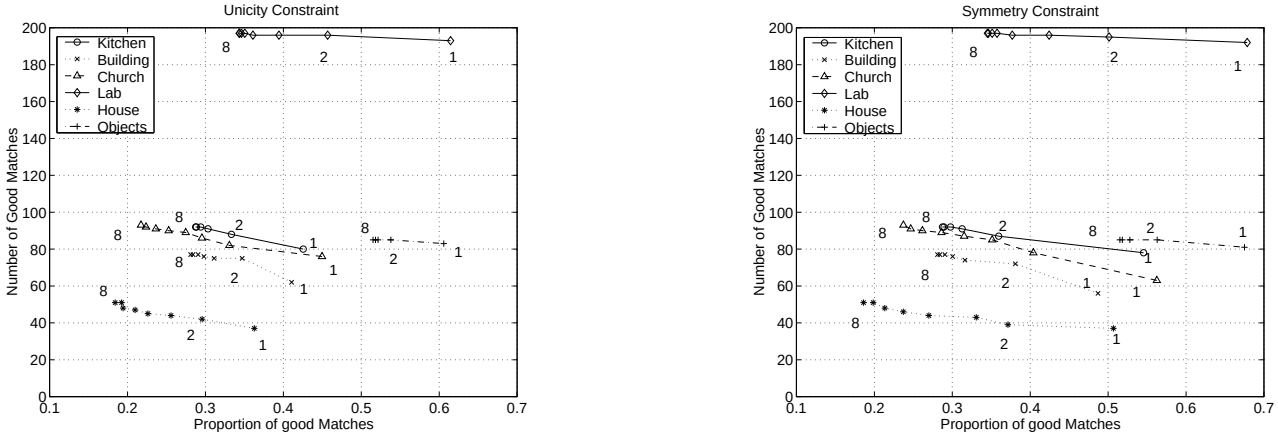
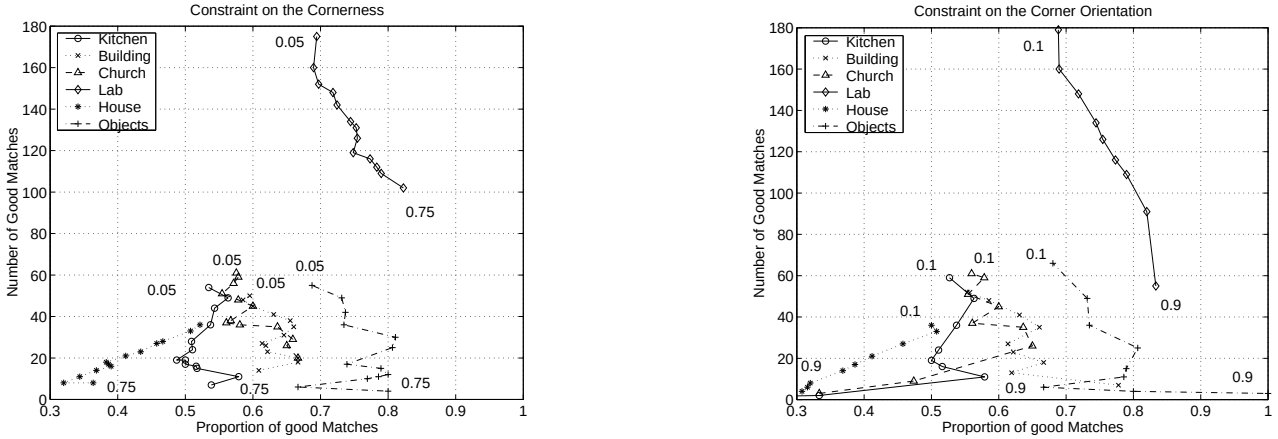


Figure 7: Correlating corners of Fig. 1 and 2. Each data point represents a test with particular parameter values. (a) ASD correlation threshold between 200 and 1100 on 11×11 windows. (b) ASD window size between 3×3 and 21×21, with ASD threshold of 700. (c) VNC threshold between 0.7 and 0.925, on 11×11 windows. (d) VNC window size between 3×3 and 21×21, with VNC threshold of 0.8.



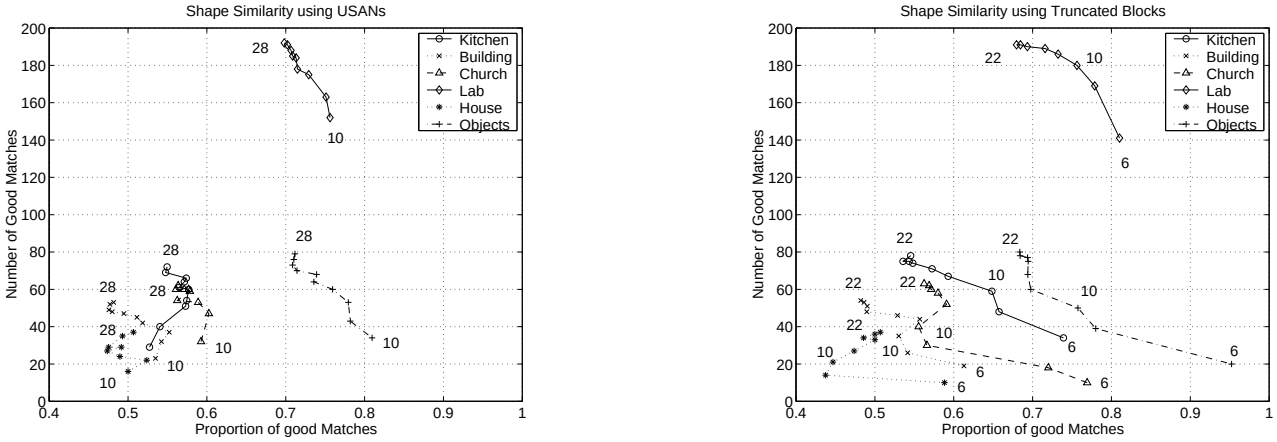
(a) (b)

Figure 8: (a) Imposing the unicity constraint with varying order while applying VNC to the corners of Figure 1 and 2. (b) Imposing symmetry on the sets obtained in (a). VNC is applied to 9×9 windows with a threshold of 0.8.



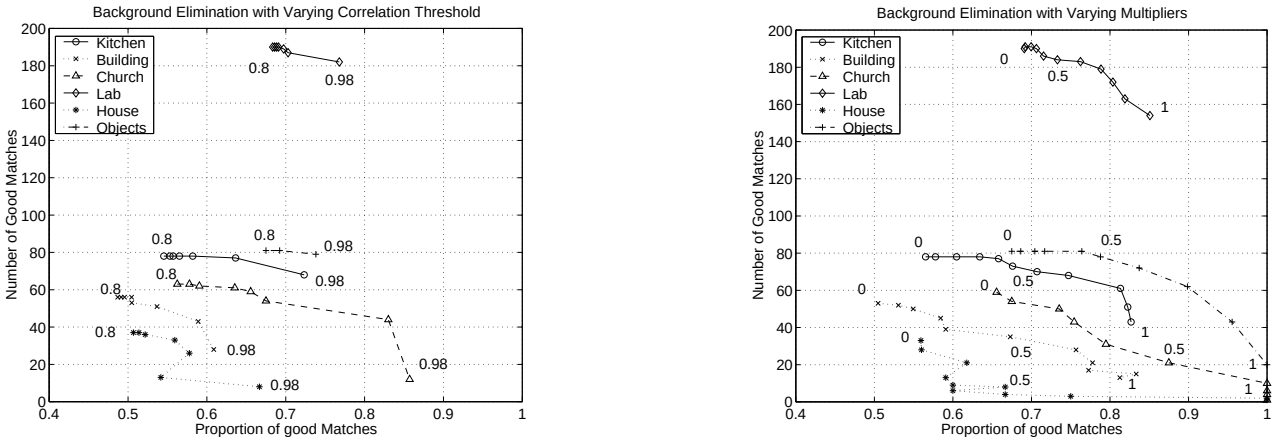
(a) (b)

Figure 9: Applying constraints on the corner characteristics. (a) Constraint on the cornerness. (b) Constraint on the corner orientation.



(a) (b)

Figure 10: Applying the shape similarity constraint on 9×9 windows, for different choices of threshold. (a) using USANs. (b) using truncated blocks.



(a) (b)

Figure 11: Applying the background elimination constraints on 21×21 windows. (a) with a varying threshold, and a multiplier of 0 for the background. (b) with a varying multiplier for the background, and a correlation threshold of 0.92.

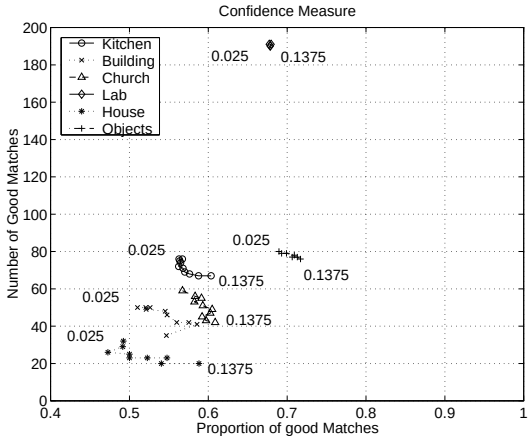


Figure 12: Constraints on the confidence measure, with varying threshold.

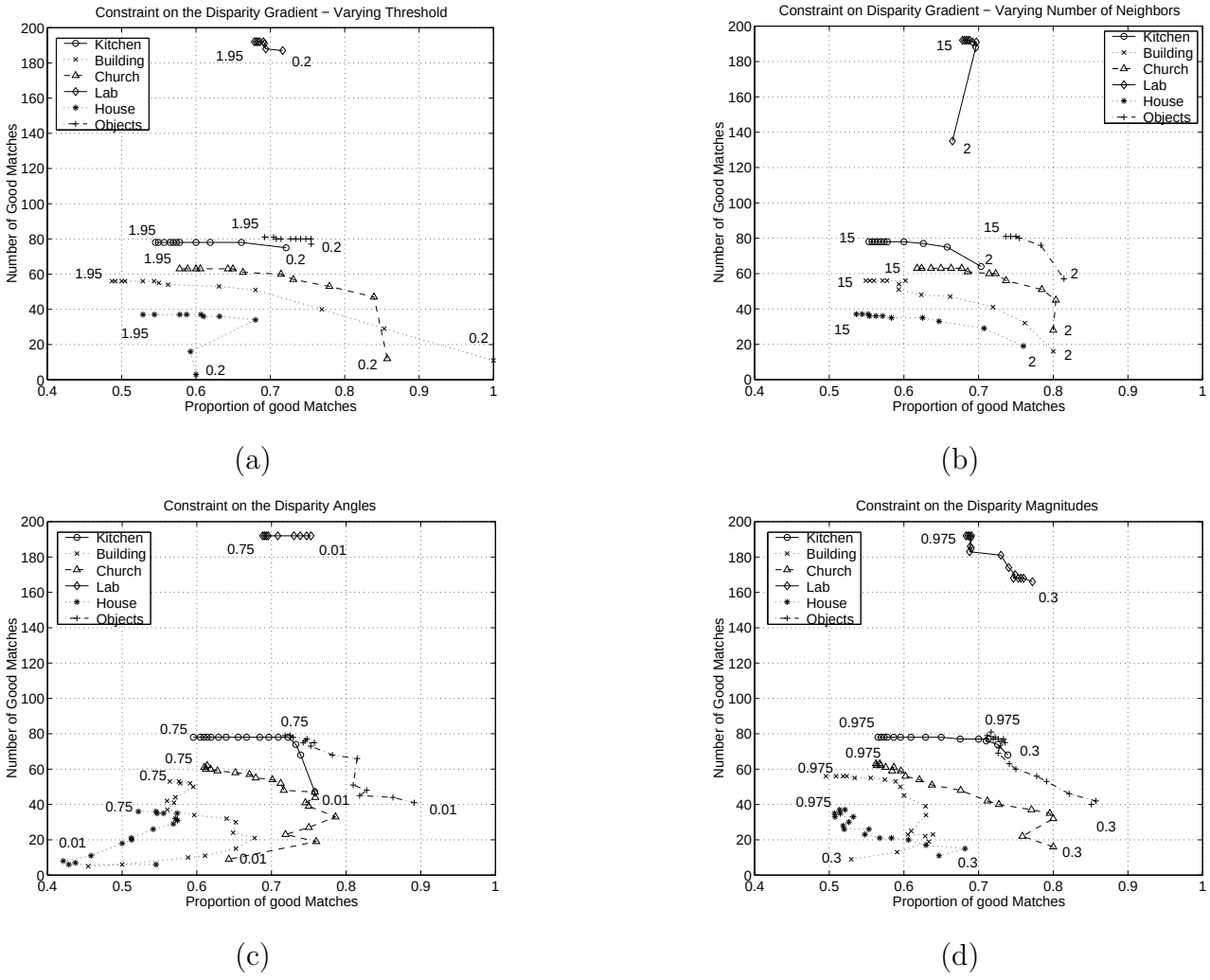
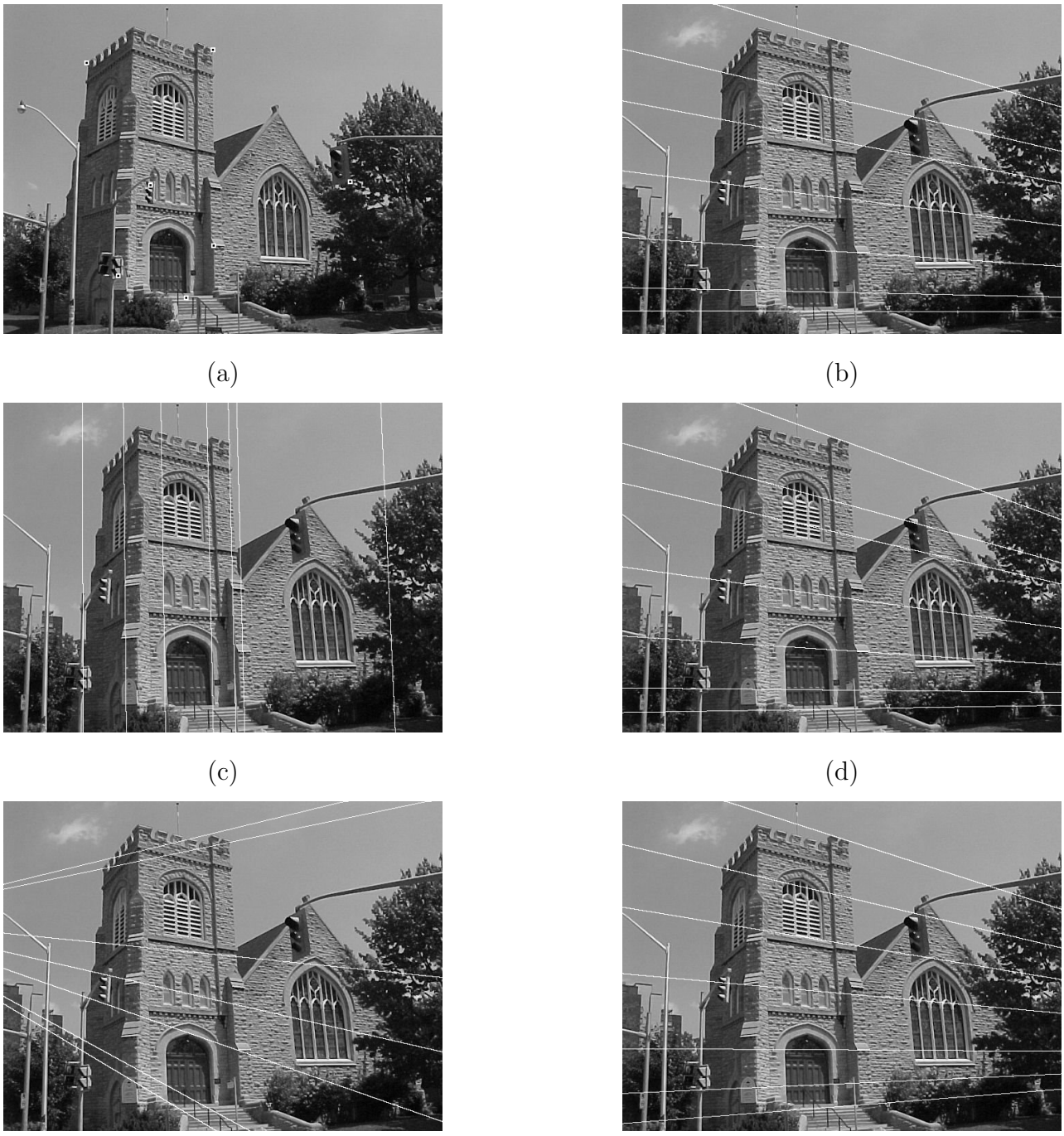


Figure 13: Constraints on the disparity. (a) Constraint on the disparity gradient, with varying threshold. (b) Varying the number of neighbors for the disparity gradient. (c) Constraint on the disparity angles, with varying threshold. (d) Constraint on the disparity magnitude, with varying threshold.



(a) (b) (c) (d) (e) (f)

Figure 14: The epipolar geometry given by fundamental matrices computed from different match sets. (a) The left image with selected points. The corresponding epipolar lines in the right image, as calculated from : (a) the set of exact matches. (c) the non-filtered match set after 500 RANSAC iterations. (d) the filtered match set after 500 RANSAC iterations. (e) the non-filtered match set after 7500 RANSAC iterations. (f) the non-filtered match set after 10000 RANSAC iterations.

References

[1] P. Anandan, Computing Dense Displacement Fields with Confidence Measures in Scenes Containing Occlusion, *Intel. Robots and Comp. Vision* vol 521, 1984.

[2] S. Arbouche, Feature Point Correspondences: A Matching Constraints Survey, *M.C.S. Thesis*, University of Ottawa, 1999

[3] A. Baumberg, Reliable Feature Matching Across Widely Seperated Views, *CVPR*, 2000.

[4] D. Bhat, S. Nayar, Ordinal Measures for Image Correspondence, *PAMI* vol. 20, no. 4, 1998.

[5] T. Darell, A Radial Cumulative Similarity Transform for Robust Image Correspondence, *CVPR*, pp. 656-662, 1998.

[6] E. Delp, O. Mitchell, Image Compression Using Block Truncation Coding, *IEEE Trans. on Communication* vol. 27, no. 9, pp. 1335-1342, 1979.

[7] R. Deriche, Z. Zhang, Q.-T. Luong, O. Faugeras, Robust Recovery of the Epipolar Gepmetry for an Uncalibrated Stereo Rig, *ECCV*, pp. 567-576, 1994.

[8] O.D. Faugeras, P. Fua, B. Hotz, R. Ma, L. Robert, M. Thonnat, Z. Zhang, Quantitative and Qualitative Comparison of some Area and Feature-based Stereo Algorithms, *Workshop on Robust Computer Vision*, pp. 1-26, 1992.

[9] M. Fischler, R. Bolles, Random Sample Concensus: A Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography. *Comm. ACM*, Vol. 24, pp. 381-395, 1981.

[10] P. Fua, Combining Stereo and Monocular Information to Compute Dense Depth Maps that Preserve Depth Discontinuities. *ijcai*, 1991.

[11] E. Guelch, Result of Test on Image Matching of ISPR wg iii, *Int. Archives of Photogrammetry and Remote Sensing*, 27 (III), pp. 254-271, 1988.

[12] C. Harris, Determination of ego-motion from matched points, *Proc. Alvey Vision Conf.*, 1987.

[13] C. Harris, M. Stephens, A Combined Corner and Edge Detector, *Alvey Vision Conf.*, pp. 147-151, 1988.

[14] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, *Cambridge University Press*, 2000.

[15] Y. Hsieh, D. McKeown, F. Perlant, Performance Evaluation of Scene Registration and Stereo Matching for Cartographic Feature Extraction, *PAMI*, vol. 14, pp. 214-237, 1992.

[16] R. Horaud, O. Monga, Vision par ordinateur - outils fondamentaux, Hermes, 1995.

[17] X. Hu, N. Ahuja, Matching Point Features with Ordered Geometric, Rigidity, and Disparity Constraints. *PAMI*, vol. 16, no. 10, pp. 1041-1049, 1994.

[18] G. Jones, Constraint, Optimization, and Hierarchy: Reviewing Stereoscopic Correspondence of Complex Features, *Computer Vision and Image Understanding*, vol. 65, no. 1, pp. 57-58, 1997.

[19] I.-K. Jung, S. Lacroix, A Robust Interest Points Matching Algorithm, *ICCV*, pp. 538-543, 2001.

[20] R. Klette, K.Schluns, A. Koschan, Computer Vision: three-dimensional data from images. Sprigner, 1996.

[21] A. Koschan, A Framework for Area Based and Feature Based Stereo Vision. *MGV*, vol. 2, no. 4, 1993.

[22] R. Laganière, A Morphological Operator for Corner Detection, *Pattern Recognition*, pp. 1643-1652, 1998.

[23] D. Lowe, Object Recognition from Local Scale-Invariant Features, *ICCV*, 1999.

[24] H. Nishihara, T. Poggio, Stereo vision for Robotics, *ISRR*, 1983.

[25] J. Noble, Finding corners, *Image and Vision Computing***6**, pp. 121-128, 1988.

[26] G. Roth, A. Whitehead, Using Projective vision to find Camera Positions in an Image Sequence, *Proc. of Vision Interface*, pp.225-232, 2000.

[27] D. Scharstein, R. Szeliski, R. Zabih, A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms, *CVPR, Workshop on Stereo and Multi-Baseline Vision*, 2001.

[28] C. Schmid, R. Mohr, Local Grayvalue Invariants for Image Retrieval, *PAMI*, pp. 530-535, 1997.

[29] C. Schmid, R. Mohr, C. Bauckhage, Comparing and Evaluating Interest Points, *ICCV*, pp. 230-235, 1998.

[30] S.M. Smith, J.M. Brady, SUSAN - A New Approach to Low Level Image Processing *Technical Report*, Oxford University, 1995.

[31] P. Torr, A. Zisserman, Robust Computation and Parameterization of Multiple View Relations, *ICCV*, pp. 727-732, 1998.

[32] T. Tuytelaars, L. Van Gool, L. D'haene, R. Koch, Matching of Affinely Invariant Regions for Visual Servoing, *ICRA*, pp. 1601-1606, 1999.

[33] Z. Zhang, Le problème de la mise en correspondence: L'etat de l'art, *Technical Report*, INRIA, 1993.

[34] Z. Zhang, R. Deriche, O. Faugeras, Q.-T. Luong, A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geomerty, *Technical Report*, INRIA, 1994.