# Chapter 5

# Uncalibrated Stereo Facet Reconstruction

*In this chapter, existing stereo matching techniques are reviewed, and the interaction between matching, reconstruction, and the epipolar geometry is discussed. An algorithm is presented for matching line segments in weakly calibrated stereo and organising them into planar facets for grasp planning.*

## 5.1 Introduction

Our goal in this chapter is to reconstruct, in an image-based frame, the shapes of objects in the robot's workspace. Many robot grippers consist of two parallel jaws, and such a mechanism is well suited to grasping objects with parallel planar surfaces. Thus a useful representation of the object for grasp planning would be a description of the visible planar facets. We shall assume that these facets are bounded by straight edges: thus the problem becomes one of matching edges in stereo views and recovering a description of the position and extent of each facet.

Our stereo vision system is weakly calibrated, meaning that the epipolar geometry and camera parameters are known only to a low level of accuracy, because only a simple self-calibration process has been performed, and the cameras may be subject to disturbances.

A large number of algorithms exist for stereo correspondence: they are summarised in section 5.2. None of these was entirely suitable for our purposes. Correlation and corner-based systems can be used to recover both structure and epipolar geometry, but are unable to match some indoor scenes in which corners are sparse, similar in appearance to one another, or confined to a few planes. Edgel-based systems give excellent results with rectified images, but are sensitive to errors in epipolar geometry because they match along epipolar lines. Line segment matching is more robust in weakly calibrated stereo, but the 3-D reconstruction of lines can be inaccurate unless extra constraints (such as junctions and coplanarity) are taken into account.

A novel system was therefore developed, extending existing algorithms for line segment matching and incorporating image-based coplanarity grouping, to reconstruct scenes composed mainly of straight edges and planar facets.

## 5.2 Review of stereo matching techniques

To reconstruct a scene from a stereo pair, it is necessary to find which points in the two views correspond to the same point in space. This is known as the *stereo correspondence problem*. For object reconstruction or recognition, the matched features must then be *grouped* to form surfaces and objects.

### 5.2.1 Feature extraction

Correspondence algorithms (reviewed in [98]) operate either on individual pixels or general patches of the images, or more commonly on a smaller set of *features* extracted independently in each image (see figure 5.1):

**Intensity-based matching.** In some cases, pixels on corresponding epipolar lines in the two images can be matched by their intensities [104], but this is easily defeated by noise, reflectance characteristics or differences in the photometric response of the cameras. To overcome noise and camera response differences, the outputs of local filters are considered, such as the smoothed derivative of intensity along the epipolar lines, or the ratio of intensity derivative to intensity [126].

**Cross-correlation.** Patches between images can be matched by looking for maxima of *normalised cross-correlation* or some other measure of similarity between the views [91]. This assumes that the apparent motion of each patch between views is a translation, i.e. depth variations are small.

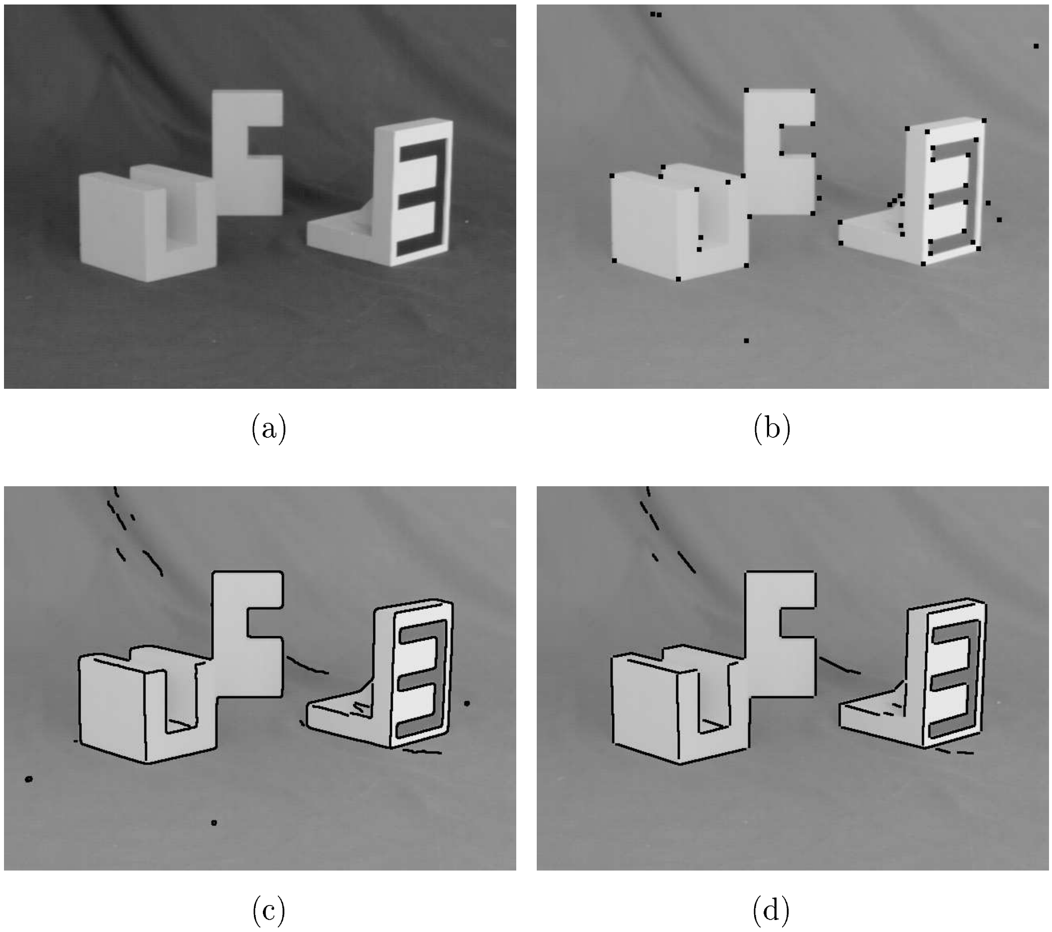


Figure 5.1: Features extracted from an image: (a) original image; (b) corners detected using INRIA corner finder [138]; (c) edges detected using Canny's algorithm [14]; (d) straight line segments fitted to edges.

**Edgels.** These are points of maximal intensity gradient after smoothing with a filter designed to reject noise [14, 30], or 'zero crossings' after convolving with $\nabla^2\mathbf{G}$ at a given scale [81, 45]. This provides a reasonable distribution of matchable features across the images which are well localised and also geometrically very significant, often coinciding with depth or orientation discontinuities [100, 32].

**Corners.** These are points around which intensity variation occurs in more than one direction, making them good candidates for matching by cross-correlation as well as points of likely geometrical significance. They can be detected using ratios of first and second-order differential operators to find edge-like points having maximal curvature [94, 132], or points of maximal auto-correlation after Gaussian smoothing [57]. By matching only the corners, the complexity of the correspondence problem is greatly reduced; the rest of the scene can be reconstructed by interpolating between corners using triangulation [55].

**Line and curve segments.** Edgels generally exhibit continuity and are grouped into *chains* in each image. These chains can then be segmented into straight line segments [6] or parametric curves such as B-splines [47, 17], and entire segments matched between images, reducing the computational complexity. In the presence of noise and quantisation errors, lines and parametric curves can be localised to higher precision than individual pixels. However, the segmentation of chains is not always stable with respect to viewpoint changes, so the matches between images are not always 1-to-1 [84].

**Higher-level features and groupings.** In some environments in which the forms of objects are modelled it is possible to apply 'perceptual grouping' to features, organising them into higher-level structures by the detection of symmetry, parallelism, clustering or similarity within the image [80, 74]. By matching entire groupings, the search space for correspondence is reduced. Systems have been proposed which match a hierarchy of features, using both bottom-up monocular grouping and top-down stereo matching [87, 20].

### 5.2.2 Matching constraints

Most stereo systems exploit the *epipolar constraint* which restricts the search for matching features to a one-dimensional one. Often they require the images to be rectified. Limits may be imposed on the magnitude of the horizontal *disparity* between views, effectively bounding the depth range of reconstructed features.

A pair of features lying within the allowed window of disparity are considered a candidate match if they are sufficiently similar in appearance (e.g. for edgels, if their orientation difference is within a certain range). However, ambiguities often arise, which can be resolved only after considering the interactions between candidate matches. Constraints used to disambiguate matches include:

**Uniqueness.** A point in the world has only one 3-D position at a time. Therefore a feature in one image can match at most one feature in the other [82]. This constraint can be broken when matching group features such as curve segments which may be organised differently in each image [84, 6].

**Ordering.** The order of matching features along the epipolar lines will usually be the same in both images [7]. This constraint is occasionally broken at the *occluding edges* of slender objects, or where there is transparency [98, 60], but is obeyed by most images of opaque solid objects.

**Surface shape.** Constraints can be imposed on the shape of reconstructed or interpolated surfaces, to aid matching. The simplest of these are *smoothness* constraints, which assume local planar structure [60, 83]; and limits on the *disparity gradient*, to favour a continuous variation of depth [100]. These constraints cannot be applied at occlusion boundaries in the images.

**Continuity.** When matching edgels, it can be assumed that edgels that are continuous in the image also connect in space, and that the disparity of an edge will change smoothly along its length as it crosses the epipolar lines [98, 95]. Similarly line or curve segments which meet at a point in one image are likely to correspond to segments which meet in the other [101].

### 5.2.3 Matching algorithms

Many algorithms have been proposed for binary matching for computer vision and other applications. The problem in general is to find the subset of the candidate matches (of which there are up to $n^2$ where $n$ is the number of features) which give an optimal correspondence between images, subject to given matching constraints. To a large extent, it is the form of those constraints which determine the algorithm used and its complexity.

With only the uniqueness constraint, a procedure such as the 'stable marriage' algorithm of Knuth et al. [66] can recover the optimal set of matches. This iterates through the features of one image, enumerating the candidate matches for that feature and choosing the one with greatest strength, whose feature in the other image is not already associated with a stronger match. A record is kept of the best match found so far for each feature. Complexity is $\mathcal{O}(n^2)$ in the number of features. Where additional mutual exclusivities must be imposed, a further depth of iteration is required to find all the matches, and complexity rises to $\mathcal{O}(n^3)$.

For edgel matching, correspondence can be formulated as a dynamic programming problem in which a 'path' must be found across each epipolar plane obeying the ordering and uniqueness constraints whilst seeking to minimise the disparity gradient, visiting each accepted match in left-to-right order [7]. Complexity is $\mathcal{O}(n^3)$ in the number of edgels on each epipolar line. Results can be enhanced by using connectivity information from the neighbouring epipolar planes [95], but this greatly increases computational complexity.

Correlation-based region matching can be made more efficient by imposing surface shape smoothness constraints and by the use of multi-scale algorithms that estimate disparities at successively finer resolutions [93]. Such an approach has also been applied to edge matching using a bank of $\nabla^2\mathbf{G}$ filters of different sizes [44].

In general, stereo matching of discrete features can be posed as a *cooperative* problem, with matching constraints taking the form of mutual positive or negative support, and/or mutual exclusivities between sets of two or more matches. Such a problem can be solved by a *relaxation* algorithm [82, 100, 84, 20]. Essentially, this iteratively updates the support given to each candidate match so as to minimise an objective function encoding the matching constraints, until all matches have been selected or rejected. An algorithm of this class will be used in section 5.4 in the design of our weakly-calibrated matching system.

The general form of a stereo vision system is summarised in figure 5.2.

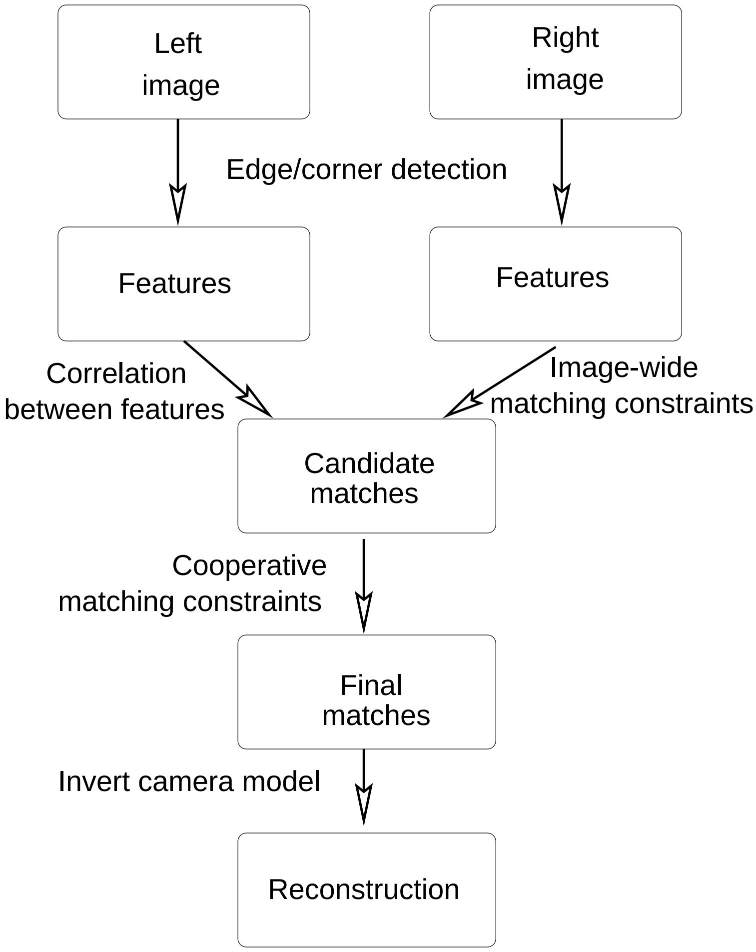



Figure 5.2: Overview of the steps and data representations of a typical feature-based stereo matching algorithm.

## 5.3 Uncalibrated stereo considerations

The behaviour of stereo matching systems with *uncalibrated* cameras is now discussed. Corner-based matching can be used to recover both scene structure and epipolar geometry, provided there is a sufficient density of correct matches. Edge-based matching is denser and more robust, but *cannot* be used to update the epipolar geometry directly. Hence reconstruction is sensitive to rectification errors, and uncalibrated reconstruction is not generally possible. Coplanarity constraints are one way to resolve this problem, and groups of coplanar features can be identified in uncalibrated stereo and reconstructed up to an affinity.

### 5.3.1 Point features

Much recent work on reconstruction without a prior epipolar constraint has relied on point features such as corners. These can be matched in stereo [138] and successfully tracked over long sequences of images in structure-from-motion [10], especially in natural scenes which tend to be rich in non-repeating texture. Robust statistical methods such as RANSAC [38] enable the fundamental matrix to be estimated even when there is a proportion of false matches [138, 127, 70]. The epipolar constraint enables most of the erroneous matches to be rejected, and the scene reconstructed up to a projectivity or affinity (as in chapter 2).

Figure 5.3 shows the results of the uncalibrated corner matching algorithm of Zhang et al. [138] on some indoor test scenes. The system uses cross-correlation between views to match corners, and a relaxation algorithm to enforce the uniqueness constraint; a fundamental matrix is then fitted to the correspondences using a robust estimator (Least Median of Squares [110]), and the matching process repeated using the recovered epipolar constraint.

This algorithm works well on highly textured or heterogeneous scenes (such as the `lab` images in figure 5.3c) where many corners can be localised and matched by cross-correlation, correctly recovering both the epipolar geometry and a dense set of correspondences. However, with simpler images (such as figure 5.3d), the density of correctly matched points is lower. The ‘corners’ detected in the images do not always coincide with polyhedral corners. Thus in the absence of texture, corners alone may not give a sufficiently detailed reconstruction for surface modelling, e.g. for robot grasp planning.

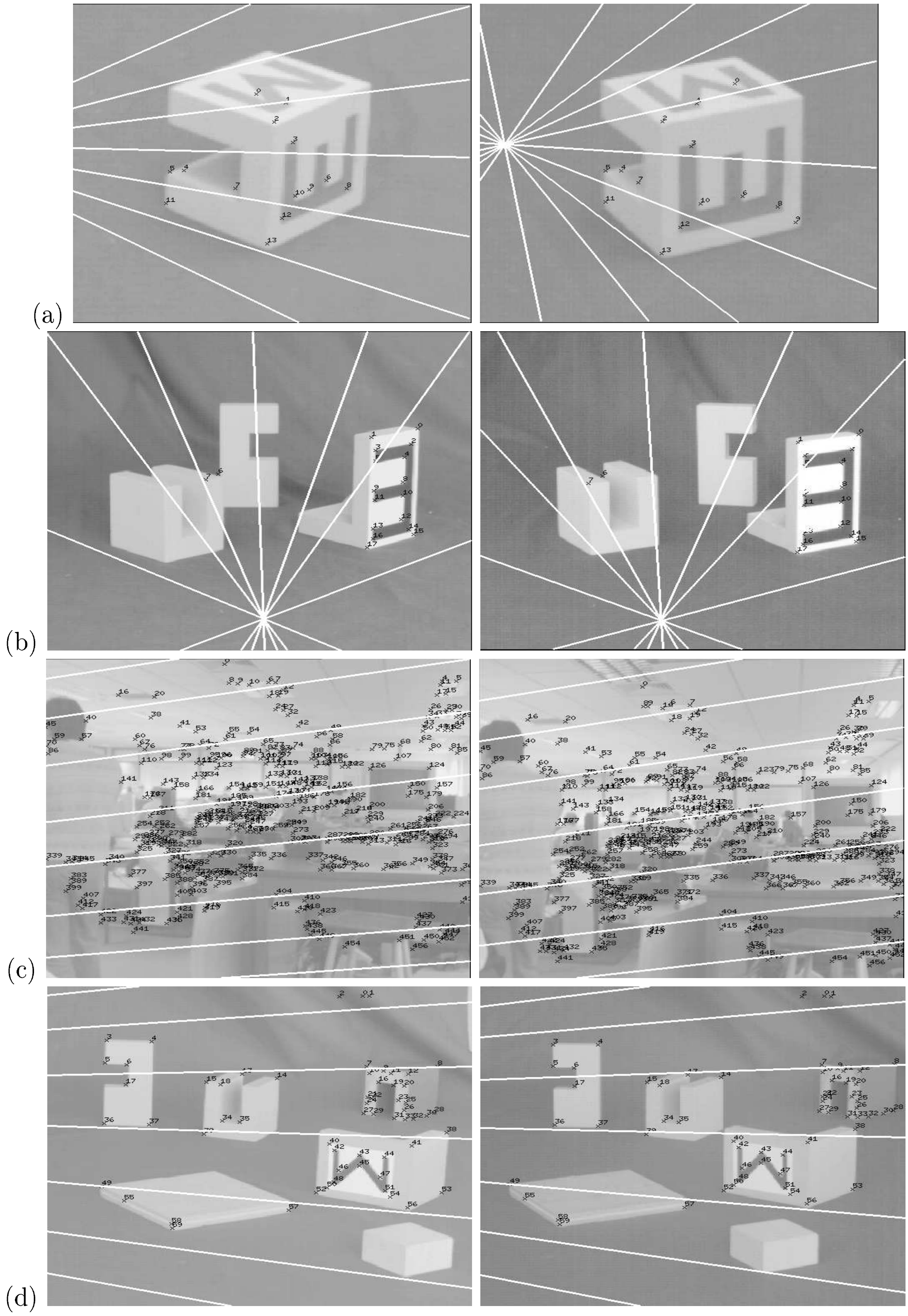


Figure 5.3: Results of corner matching [138] and estimated epipolar lines: (a) `cube` scene; (b) `test` scene; (c) `lab` scene; (d) `blocks` scene. Sparsity of corners and high incidence of coplanarity leads to incorrect solutions in (a) and (b).

On some images, such as the `test` and `cube` pairs, uncalibrated corner matching fails entirely: the corners look too similar to be distinguished by correlation, and the predominance of planar facets proves to be a hindrance rather than a help: large coplanar subsets of points can defeat the robust estimator and lead to a degenerate solution for the fundamental matrix (figure 5.3 a, b).[1]

### 5.3.2 Edge-based features

We shall now consider the feature type most prominent in many indoor scenes, *edges*. Because they are localised in only one dimension, matching of edge elements depends upon prior estimation of the epipolar constraint, and edgels tangent to the epipolar lines cannot be matched uniquely [100]. This is the *aperture problem* in stereo [131]. But by grouping the edgels into line or curve segments, complexity is reduced and the epipolar constraint can be relaxed to require that matching segments *overlap* when projected into the rectified vertical axis (that is, their ranges of $y$ and $y'$ values intersect), the degree of overlap indicating how well aligned the segments are in the two views [6, 137]. This allows some segments to be matched with only approximate epipolar calibration.

Ayache [6] presents an algorithm for matching line segments in rectified stereo pairs, in which matches give support to each other if they are nearby (according to a coordinate bucketing scheme) in both views. Hypothetical matches are formed between line segments if the $y$ coordinate of the midpoint in one image falls within the range of $y'$ values of the other (and *vice versa*), and if their lengths and orientations are similar within limits inferred from the camera geometry [4]. Matches are accepted if they give rise to maximal cliques of supporting matches under the uniqueness constraint. Reconstruction is performed under the assumption that the cameras are accurately calibrated.

Zhang [137] has used a numerical search method to solve for the camera motion (hence the epipolar constraint), given two intrinsically-calibrated images of matched line segments, by seeking to maximise the total epipolar overlap, thus permitting *uncalibrated* reconstruction. However, the system is not well constrained, and a very large number of segments is required for a satisfactory solution.

[1] One way to avoid this problem would be to search for sets of points consistent with a degenerate model (e.g. planes), then solve for the epipolar constraint using whatever features (if any) remain. This is essentially the approach taken by the PLUNDER motion segmentation algorithm [127].

### 5.3.3 The problem with vertical disparity

Consider the case when the epipolar constraint is inaccurately modelled. Corresponding features will exhibit misalignment or *vertical disparity*.[2] This vertical disparity will affect the 3-D reconstruction of any one-dimensional features such as edgel chains, lines and curves, which depend on the epipolar geometry to establish matching points and recover depth. What is more, the error varies with orientation: each unit of offset induces a horizontal disparity error of $\tan\theta$, where $\theta$ is the angle to the vertical (horizontal lines cannot be reconstructed at all). Figure 5.4 shows the results of this error, and its disruptive effect on plane reconstruction.

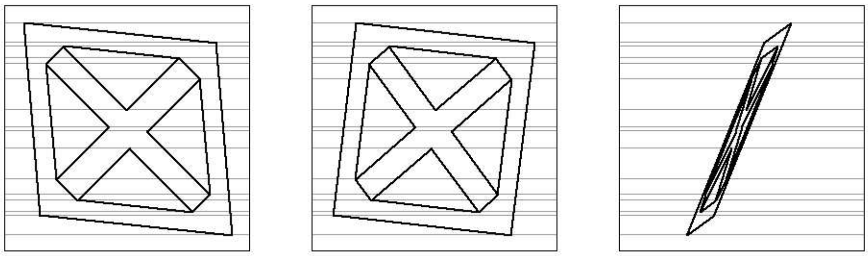
The vertical disparity problem can be overcome if line or curve segments are known to have matching endpoints: one can simply solve for the 3-D coordinates of the endpoints in some approximate (affine) world frame, or use them to re-estimate the fundamental matrix. This is not always the case in real images, due to the fragility of line fitting and segmentation (in the case of curve segments, it may be possible to match other distinguished points such as bitangencies for plane curves, or tangencies to the epipolar lines [102, 5]). However, where line segments meet at a junction in space — and more generally where they are coplanar but not parallel — their *intersections* in two images can be used as accurate point correspondences; this is exploited in section 5.5.

### 5.3.4 Coplanarity grouping of line segments

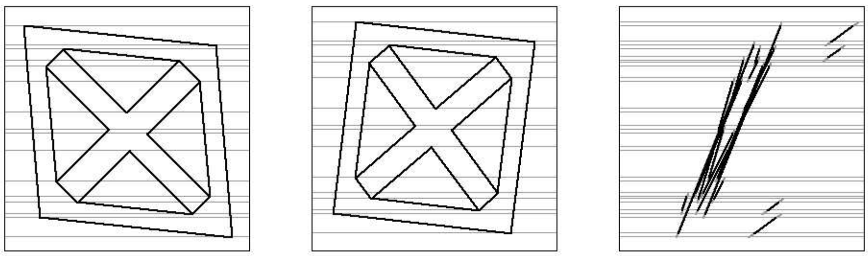
For the reconstruction of planar surfaces, it is necessary to group matching line segments into planar facets. Plane grouping could take place between matching and reconstruction; it could also be performed concurrently with the matching process, to favour candidate matches which belong to well-supported planar facets.

Coplanarity of line segments cannot be detected unambiguously in a single image, but requires stereo. Lines that are parallel in two weak perspective images are necessarily coplanar in space, though they might not be physically connected. Line segments which meet at a junction in both images are generally coplanar, unless the junction has been caused by occlusion (a 'broken T-junction').

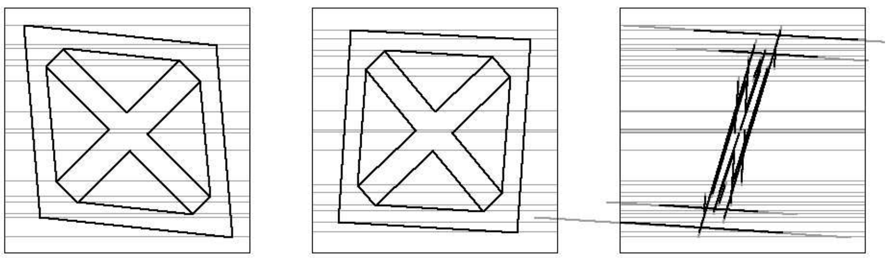
[2] The terms *horizontal*, *vertical* and *disparity* are used here to mean the $x$, $y$ and $(x' - x)$ quantities in a rectified coordinate system. This does not imply that both images must be rectified, merely that some estimate of the epipolar constraint is available by which rectified coordinates could be calculated.



(a) A pair of synthetic images of 20 line segments,
and side view reconstructed from horizontal disparities



(b) The same images offset vertically by about 2% of the image height



(c) One of the images has been rotated by 3°,
causing non-uniform vertical disparity

Figure 5.4: The effect of vertical disparity on edgel-chain, line or curve segment reconstruction in rectified stereo. Because the error induces a horizontal disparity that varies with orientation, the planar structure of the scene is destroyed.

### With known epipolar constraint

Coplanarity can be tested using the following theorem: **Two non-parallel lines are coplanar if and only if their intersections in two images lie on corresponding epipolar lines.**[3] Whereas if they are not coplanar, there will be a vertical offset between their apparent intersections in the two images (in fact, this offset or *pseudo-disparity* is a cue to the depth difference of the segments [78]). This test is sensitive to errors in the epipolar geometry, though appropriate limits can be placed on the amount of vertical disparity allowed. Its resolution is therefore degraded as epipolar constraint uncertainty increases.

### With unknown epipolar constraint

Even without camera calibration or an epipolar constraint, a pair of images can be segmented into planar regions by the following theorem: **Two views of a planar surface are related by a two-dimensional projective transformation; features are consistent with this transformation if and only if they lie on the plane**. In weak perspective, the transformation will be affine (see chapter 2).

Faugeras and Lustman [34] use this theorem to recover correspondence of line segments on a single plane, by hypothesising a set of matches that define a collineation between views, and testing for consensus with other segments. A Kalman filter is used to refine the estimate of the transformation. They show that two views of the plane allow structure and motion to be computed up to scale and a two-fold ambiguity (for intrinsically calibrated cameras). The ambiguity can be resolved using a second plane or a third view. The principle is not tested on more complex scenes.

Sinclair and Blake [118] use 2-D projective invariants to detect sets of 5 or more coplanar points (given corner correspondences) and construct an approximate piecewise-planar model of terrain with application to mobile robot navigation.

A drawback of this method is that it requires a minimum of four (or three) lines or points to define the projective (or affine) transformation, and at least one other feature to verify it. Hamid et al. [52] show that, for a typical stereo camera geometry, feature localisation must be to sub-pixel accuracy to segment nearby planes reliably using such an algorithm alone.

[3] Outline proof: For the lines to intersect in both views at corresponding epipolar lines, each line must exhibit the same disparity where it crosses this epipolar plane, therefore they must be at the same depth. This is degenerate if one or both lines are horizontal.

## 5.4 An algorithm for uncalibrated matching

This section describes a stereo matching algorithm which operates on line segments. It is based upon existing techniques, but has been designed to deal explicitly with the bounded uncertainty in epipolar line correspondence found in weakly calibrated stereo. Figural relations between segments are analysed to add robustness.

### 5.4.1 Feature extraction

It was found that corners and correlation-based algorithms do not always give the required density and resolution of reconstruction for the recovery of facets. Edgel-based matching along epipolar lines was also rejected because of the problem of epipolar misalignment, which could disrupt planar surfaces. Edge segment matching was therefore chosen. For simplicity only straight edges are considered here; though much of our approach could be extended to curve segments.

Line segments were extracted from Canny edgel data [14] using a recursive algorithm that searches for straight segments of edgel chains (figure 5.1(d), p73). Each segment is represented by its endpoint coordinates; and associated with it are uncertainty measures for its orientation and normal offset, obtained from the residual errors after fitting it to the edgels by orthogonal least squares [122].

### 5.4.2 Monocular relations

Consider figural relations between just two line segments in a single image (figure 5.5). It is assumed that the edges are extracted in such a way that segments do not cross. Notable binary relations include:

**Parallelism.** Parallelism between segments (within some given margin of error) can be determined very quickly using an orientation bucketing scheme. It is obviously not meaningful to look for the intersection of parallel lines.

**Collinearity.** This is a special case of parallelism. Collinearity is used to extend the uniqueness constraint to line segments: because of the fragility of line segmentation, one or more collinear segments in the first image may match one or more collinear segments in the second.

**Junction.** This occurs when an endpoint of one segment in an image lies within some maximum distance of an endpoint of another segment, suggesting that the edges are coincident in space.

**Collinear junction.** This occurs when segments are collinear and meet at a junction. Such junctions are not generally stable between views; they could also be a component of a 'broken T-junction' between occluding and occluded segments.

**T-junction.** This occurs when the endpoint of one segment lies close to another segment. It suggests an occlusion boundary.

The system identifies these relations between segment pairs in each of the two images. The threshold angle for parallelism was set to $3^{\circ}$ greater than the given orientation uncertainty, to detect parallel lines in the presence of optical distortion or mild perspective effects. For junction detection, endpoints were required to be within 6 pel (or up to 12 pel for longer or more uncertain segments) of a point extended 3 pel out from the other segment's endpoint. These limits were chosen to defeat the observed 'corner-rounding' behaviour of the edge detection and line fitting code, caused by the isotropic smoothing stage of Canny's algorithm.

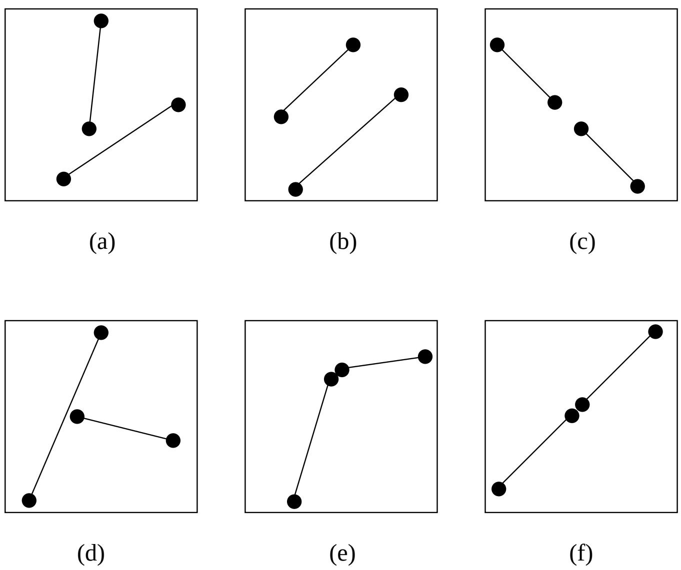



Figure 5.5: Binary figural relations between line segments in a single image: (a) general case (b) parallel (c) collinear (d) T-junction (e) junction (f) collinear junction.

The search for related segments could be accelerated considerably by sorting them into 'buckets' [65] of segments whose orientations or endpoint coordinates lie within particular intervals: for a typical image of 200 segments with 2000 related pairs, execution time is approximately 0.25 seconds.

### 5.4.3 Candidate matches

The next stage in the algorithm is the enumeration of 'candidate matches,' or segment pairings between images that could possibly be images of the same edge in the world. To generate these, an approximate epipolar constraint is required — this can be a linear estimate, provided by just four reference points[4]. The matching criteria are based on those of Medioni and Nevatia [84] and Ayache [6], with some adaptations for weakly calibrated stereo:

- **Epipolar overlap** as a fraction of total vertical extent (min. 25%),
- **Length ratio** in a 'vertically stretched' rectified frame (max. 3),
- **Orientation difference** in the rectified frame (max. 60°),
- **Disparity limits** extrapolated from the disparities of the reference points.

The constant values specified in the criteria were chosen by hand to optimise performance, though the system is not critically sensitive to any of them.

Notes:

1. The epipolar overlap criterion is broadened somewhat by allowing up to 16 pel of vertical offset, to overcome rectification errors. This allows many nearly-aligned segments to be matched (including horizontal ones) even when they show no overlap at all.

2. Length and orientation comparisons are performed in a 'vertically stretched' rectified coordinate system. This gives more weight to the direction normal to the estimated epipolar lines which should be invariant to viewpoint changes; but also takes into account the component along the epipolar lines, to allow near-horizontal features to be compared and matched. For typical stereo camera configurations, length and orientation differences will be within the above bounds for all except the most foreshortened of matching segments.

[4] If no reference points are available, it is assumed that the epipolar lines are approximately horizontal and the range of permitted disparities is ±100 pel.

**Angle threshold and orientation-based support:**

$$a_{ij} = \begin{cases} 2(\cos\theta_{ij} - 0.5) & \text{if } \cos\theta_{ij} > 0.5, \\ 0 & \text{otherwise.} \end{cases}$$

**Length-ratio threshold and support:**

$$b_{ij} = \begin{cases} 1.5(L_i/L'_j - 0.33) & \text{if } L_i < L'_j < 3L_i, \\ 1.5(L'_j/L_i - 0.33) & \text{if } L'_j < L_i < 3L'_j, \\ 0 & \text{otherwise.} \end{cases}$$

**Epipolar overlap constraint and support:**

$$\lambda_{ij} = \frac{\textsc{overlap}(y_{i[0]}..y_{i[1]}; y'_{j[0]}..y'_{j[1]}) + 16.0}{\textsc{max}(y_{i[0]}, y_{i[1]}, y'_{j[0]}, y'_{j[1]}) - \textsc{min}(y_{i[0]}, y_{i[1]}, y'_{j[0]}, y'_{j[1]})}$$

$$c_{ij} = \begin{cases} \frac{4}{3}(\lambda_{ij} - 0.25) & \text{if } 0.25 < \lambda_{ij} < 1, \\ 1 & \text{if } \lambda_{ij} > 1, \\ 0 & \text{otherwise.} \end{cases}$$

**Midpoint disparity limits:**

$$\delta_{ij} = \tfrac{1}{2}(x'_{j[0]} + x'_{j[1]} - x_{i[0]} - x_{i[1]})$$

$$d_{ij} = \begin{cases} 1 & \text{if } \delta_{MIN} < \delta_{ij} < \delta_{MAX} \\ 0 & \text{otherwise.} \end{cases}$$

**Overall intrinsic support of the candidate match:**

$$\sigma_{ij} = a_{ij}\, b_{ij}\, c_{ij}\, d_{ij}$$

Table 5.1: Algebraic summary of the constraints for a candidate match between segments. $i$ and $j$ are segment numbers in the first and second image respectively. Angles and lengths are in a ‘stretched’ rectified frame in which the vertical component has double weight. The pair is considered a candidate match if $\sigma_{ij} > 0$.

3. Matches are only permitted on segments having the same sense in each image, i.e. originating from edges having the same sign of gradient. Practically all correct matches will meet this criterion (except occasionally at an occlusion boundary where there is background intensity variation), whilst a further 50% of false matches are rejected [6].

Again, bucketing is used to speed up the search, indexing the segments which intersect a number of epipolar bands. Each test assigns a number to the candidate match, indicating by what margin the criterion is met; these are multiplied to yield the *intrinsic support* of the match (see table 5.1).

In tests with 8 stereo pairs of blocks and laboratory scenes, between 5%–20% of the line segments in each image could not be matched according to these criteria, and about 50% had more than one candidate match. The average number of candidate matches per line segment varied between 1.5 and 3.0.

### 5.4.4 Inter-match constraints

In our scheme, matching constraints are expressed by two types of relations between candidate matches: mutual exclusivities between pairs of matches (hereinafter dubbed RIVALS), and mutual positive support between matches (FRIENDS). The ordering constraint is extended for weakly calibrated stereo, and a novel connectivity constraint is introduced, based on junction relations common to both images.

**Uniqueness constraint.** Matches that share a segment in either image are RIVALS, unless they are collinear and connected in the other image (as this could be due to fragmentation of the edgel data).

**Ordering constraint.** For approximately rectified stereo views, we extend the ordering constraint to matches that 'cross over' in a vertical as well as a horizontal sense (figure 5.6). 'Epipolar ordering' is tested for segments that intersect one another's projection onto the (rectified) vertical axis in both views; 'General ordering' is violated by segments that cross one another during a linear warp between views (assuming matching endpoints), which was found by human inspection to be an important cue for spotting inconsistent matches. Out-of-order matches are RIVALS.

**Collinearity.** Matches that are collinear in *both* images are FRIENDS (since this does not generally happen by accident, it suggests they are correct matches and derive from collinear features in the world).

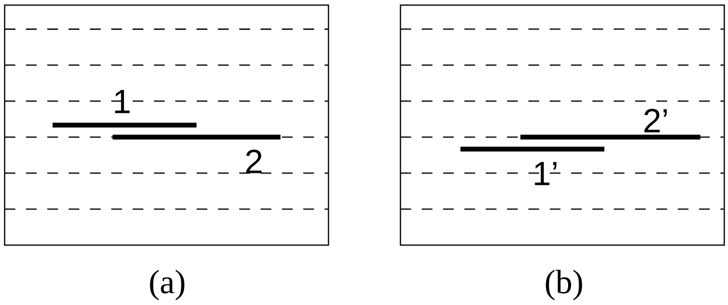



Figure 5.6: Ordering constraint violation by segments with vertical disparity: matches 1 and 2 are out of order — they 'cross over' between views (a) and (b), even though they are ordered in the direction of the estimated epipolar lines.

**Connectivity.** Matches are FRIENDS if they have *common* junction or T-junction relations in both images; but RIVALS if they have *incompatible* junction relations (i.e. they meet at different endpoints, implying a large relative motion between views).

### 5.4.5 Constraint propagation

Our goal is to try to find a set of matches with maximal total support, consistent with all the matching constraints. The solution should favour pairs matches which are FRIENDS of other chosen matches, but not include any pairs of RIVALS.

This is achieved by an iterative algorithm that propagates FRIEND and RIVAL constraints between matches. To start with, a proportion of each candidate match's intrinsic support is added to the support of each of its FRIENDS.

At each iteration, we enumerate the 'winning' matches having greater support than any of their RIVALS[5]. A proportion of the *least ambiguous* winners is then selected — these are the winners with the greatest support difference over their nearest rival. These matches are considered correct: their FRIENDS receive extra support, and their RIVALS are eliminated. As matches are eliminated, they withdraw the support that they earlier gave to their FRIENDS. The process repeats until all matches have been either promoted or eliminated.

[5] The search is made efficient by use of the uniqueness constraint which partitions the matches into small sets of RIVALS (where there are multiple collinear matches, for simplicity only one is promoted at each iteration).

This is a *'some-winners-take-all'* algorithm (similar to that described by Zhang et al. [138] in the context of corner matching; cf. [100] for a related strategy for edgel matching). In our implementation, only $\frac{1}{4}$ of the winning matches are promoted on the first iteration, as this helps to prevent early convergence to a local minimum; thereafter this rises to $\frac{2}{3}$. As with other relaxation algorithms, an optimal solution is not guaranteed, though the final set of matches must be consistent and locally optimal. Convergence is assured so long as at least one winner is promoted each time, and generally occurs after 4–10 iterations. With the use of suitable data structures, already-matched segments can be skipped, making each iteration faster than the previous one.

### 5.4.6 Epipolar constraint re-estimation

If the initial estimate of the epipolar geometry was inaccurate (or missing, and assumed horizontal), it is useful to recompute the epipolar constraint and repeat the matching process. We use the *junction* relation between segments to find intersection points that correspond between views. There are very few outliers amongst such points, but some of the junctions may be false intersections generated by occlusion ('broken T-junctions'); therefore, an iterative re-weighting scheme is used to reject those furthest from their supposed epipolar lines.[6] The parameters of the epipolar constraint (in the form of equation 2.9) are estimated using linear least squares.

Often the number and accuracy of correspondences is improved by repeating the matching process using the newly recovered epipolar constraint and disparity limits. It should be noted that the above matching algorithm can function even with some epipolar mismatch, and there is never any benefit to repeating the process more than twice.

### 5.4.7 Results

Figure 5.7 shows how the algorithm matches line segments on a pair of simple '`cube`' images by propagating qualified uniqueness, epipolar ordering, general ordering and connectivity constraints. The cameras were arranged by hand to be roughly rectified, but no calibration was performed. 39 and 40 line segments were detected in each image. Correspondence was solved for 35 segments in four iterations.

[6] The weight for each point match is proportional to $16 - |y' - y|$ and falls to zero for junctions with vertical offsets above 16. Just 4 junctions within this range are required for the method to be successful.

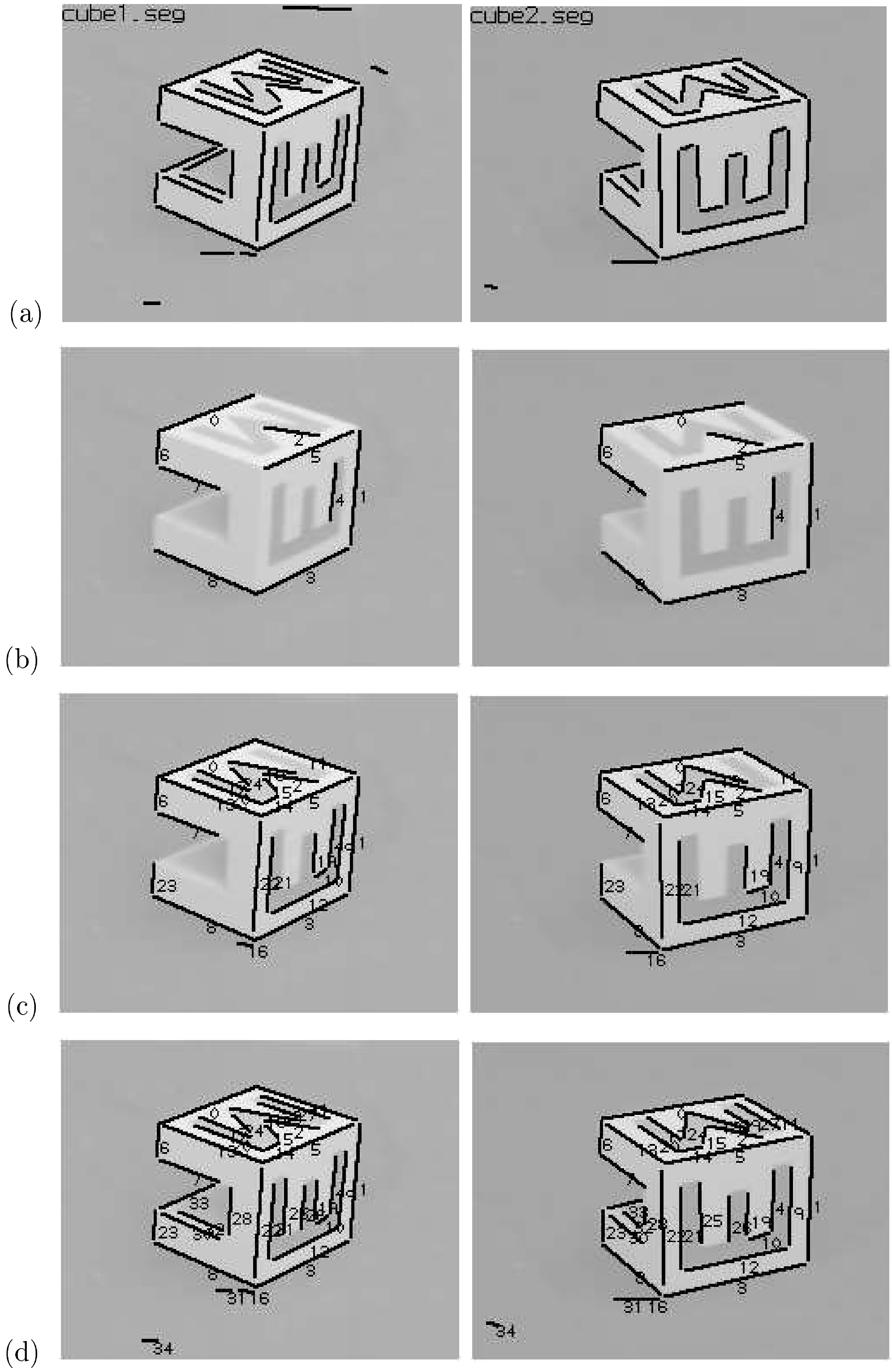



Figure 5.7: Matching by constraint propagation: (a) line segments from the `cube` images; (b) 'least ambiguous' segments matched on the first iteration; (c) segments matched after 2 iterations; (d) final result reached after 4 iterations. Uncalibrated views — epipolar lines roughly horizontal.

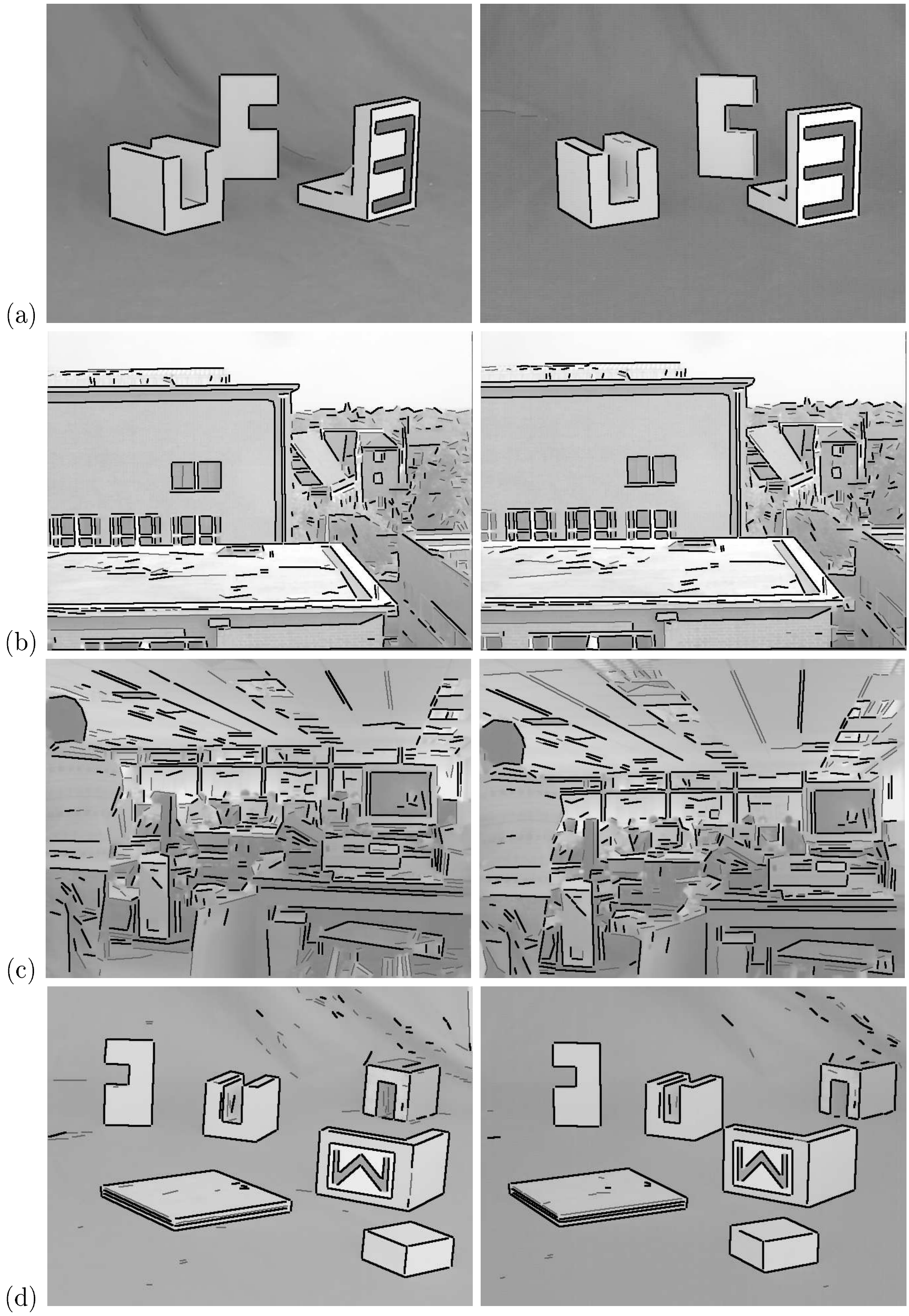



Figure 5.8: Matched (black) and unmatched (grey) segments from 4 stereo pairs: (a) `test` scene (no calibration); (b) `roof` scene (given 4 point correspondences); (c) `lab` scene (rectified using corner matching); (d) `blocks` scene (no calibration).