RECONSTRUCTION OF A ROAD BY LOCAL IMAGE MATCHES AND GLOBAL 3D OPTIMIZATION

Daniel DeMenthon
Larry S. Davis

Computer Vision Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742

Abstract

A new method is presented for reconstructing a 3D road from a single image. It finds the images of opposite points of the road; opposite points are points which face each other on the opposite sides of the road; the images of these points are called matching points. For points chosen from one side of the road image, the proposed algorithm finds all the matching point candidates on the other side, based on local properties of a road. However these solutions do not necessarily satisfy the global properties of a typical road. A dynamic programming algorithm is applied to reject the candidates which do not fit the global road.

A benchmark using synthetic roads is described, which shows that the roads reconstructed by the proposed method match the actual roads better than two other road reconstruction algorithms. Experiments with 50 road images taken by the Autonomous Land Vehicle (ALV) showed that the method is robust with real world data, and that the reconstructions are fairly consistent with road profiles obtained by fusion between range images and video images.

1 Introduction

Recent efforts in robotics have concentrated on the ability of autonomous systems to follow roads [13,12,11,5]. For robustness in a variety of conditions, these systems can be driven by a supervisor system reasoning about information provided by several algorithms, such as stereo algorithms, stereo motion algorithms, algorithms using single video frames or combining video frames and range images, Kalman predictors combining information obtained from several vehicle positions, etc. Some algorithms may monitor the road over a short distance or along a single edge, for input to a fast steering control loop [5]. Other algorithms may attempt to extend their analysis to the most distant available data in front of the vehicle, for input to longer term reasoning modules.

This paper presents a new algorithm able to reconstruct the road shape from a single image, providing the three dimensional profile of the road in front of the vehicle, often up to the point where the road becomes hidden. Reconstructing the road over a large distance presents several advantages. The reconstructions from several video frames can be overlapped, and the evidence from each reconstruction can be combined for added reliability. The vehicle can make estimations of road turns well in advance, and adjust its speed accordingly. Finally, the reconstructed road elements can be registered against the road data stored in the vehicle road map data base, providing the position of the vehicle on the map. For these reasons this long range reconstruction of the road can be usefully combined with steering control based on Kalman prediction [5].

Road reconstruction from a single image is a "shape-from-contour" problem. It is obviously under-constrained, yielding an infinity of possible road shapes unless constraints about the road structure in the 3D scene are introduced. Thus a road model has to be assumed, which provides a reasonable set of additional constraints.

The simplest model which has been applied [12] is the *Flat-Earth geometry model* ; the road is assumed planar and in the same plane which supports the vehicle, and the road image is back-projected onto this plane. The method is fast and does not deteriorate when image analysis gives ragged road image edges. But it is very sensitive to the difference between the assumed and actual camera tilt angle with respect to the ground. For a camera mounted on a vehicle at 3.5 meters above the ground, a world point at 30 meters in front of the vehicle will be reconstructed at 55 meters if the ground plane angle is overestimated by 3 degrees and at 21 meters if it is underestimated by 3 degrees, an error range of more than 100%. Consequently, the Flat-Earth algorithm is more suitable for reconstructing the road just in front of the vehicle than for a long range analysis.

More sophisticated algorithms have attempted to utilise the constraint that a road generally keeps an approximately constant width [12]. The problem with applying this constraint is that one must find the pairs of points separated by a distance equal to the road width, in straight or curved parts of the road. We call the problem of finding the correct pairs of points in the image the *matching point problem.* This is one of the problems addressed in this paper.

The constant road width constraint is not sufficient. Another constraint must be added for the reconstruction to be possible. We have chosen the *zero-bank* constraint, specifying that the road does not tilt sideways. A road model combining constant width and sero bank was originally suggested in [10].

In previous work, we developed an incremental road reconstruction method based on these constraints [2,3] in which a new pair of edge points could be found if we had already found a neighbor pair of edge points; the road edges were reconstructed incrementally from edge points close to the vehicle to edge points in the distance. This method is fragile because any increment of construction depends on the previous elements in the chain. Any failure of the road reconstruction at any point can be fatal to the further progress of the reconstruction.

This incremental method used a discrete approach. Road reconstructions based on a differential approach can be found in [7,6]. An interesting alternative to the global dynamic programming optimisation proposed in the present paper can be found in [8].

2 The matching point problem

Consider the image of a railroad track and its railroad ties, and assume that some appropriate image processing techniques have reduced the images of the rails to curves and the images of the ties to line segments between these curves (Figure 1). The positions of the end points of the tie segments on the curves of the rail are the matching points in the image. The reconstruction of the shape of the railroad track in 3D space uses the matching points and is straightforward if three hypotheses are made:

1. The width w of the railroad track is constant and known.
2. The coordinates of the vertical unit vector $\overrightarrow{V}$ are known in the camera coordinate system.
3. The railroad ties are approximately horizontal.

Obviously, the last hypothesis does not constrain the railroad itself to be horizontal. Similarly, the stairs in a spiral staircase have horizontal step edges, but the ruled surface defined by these step edges is far from horizontal.

Consider two matching points a_1 and a_2, the end points of the image of a tie. The corresponding vectors from the viewpoint O to these image points will be denoted by $\overrightarrow{a_1}$ and $\overrightarrow{a_2}$. The corresponding world points A_1 and A_2 are defined by

$$\overrightarrow{A_1} = \lambda_1 \overrightarrow{a_1},\ \overrightarrow{A_2} = \lambda_2 \overrightarrow{a_2}$$

since world points and their images are on the same line of sight.

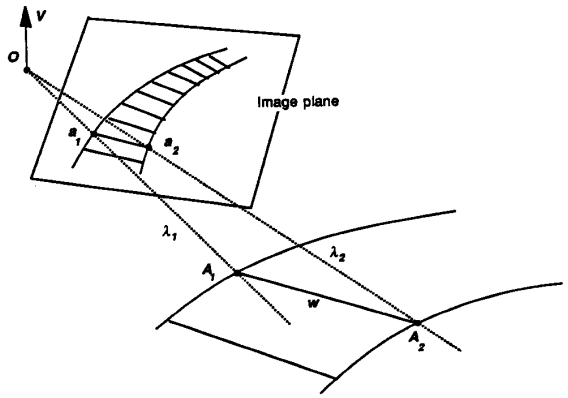

Figure 1: Reconstructing positions of world railroad ties from their images.

The world line segment is assumed horizontal; the two parameters λ_1 and λ_2 are then related by

$$\lambda_2 = m\lambda_1$$

with

$$m = \frac{\overrightarrow{a_1} \cdot \overrightarrow{V}}{\overrightarrow{a_2} \cdot \overrightarrow{V}}$$

The requirement that the distance between $\overrightarrow{A_1}$ and $\overrightarrow{A_2}$ be equal to the width w completely constraints the parameters:

$$\lambda_1 = \frac{w}{(\overrightarrow{a_1}^2 + m^2\overrightarrow{a_2}^2 - 2m\overrightarrow{a_1} \cdot \overrightarrow{a_2})^{\frac{1}{2}}} \qquad (1)$$

Thus the two curves of the rails in the scene can be in general uniquely reconstructed from their images up to a scale factor, if the ties are assumed horizontal and of constant length. Problems occur only if the railroad image crosses the horizon, as noted in [8]. In this case the ties are horizontal on the horizon line and their range cannot be determined, as can be seen from the equations above.

Consider now the problem of reconstructing a *road* from its image, once some appropriate image processing techniques have isolated the curves corresponding to the road edges in the image. This time of course we do not have the images of railroad tie segments to help us. The method we propose is thus to find as a first step the end points of line segments which correspond to images of railroad tie segments, and then do the 3D reconstruction of the end points of the images of these segments by the method just described for the railroad. We call these world segments corresponding to railroad ties "cross-segments", and their end points "opposite points". The images of these points are the "matching points". The main problem of the reconstruction of a road from its image can then be stated as: Given a point on one edge of the road image, where is the matching point on the other edge?

We choose a road model similar to the railroad model: the road is modelled as a space ribbon generated by a centerline spine and horizontal cross-segments of constant length cutting the spine at their midpoints at a normal to the spine. This modelling gives cross-segments the properties of railroad ties:

- Cross-segments are horizontal, i.e. perpendicular to the vertical (on the ALV the vertical was detected by trim sensors).
- Cross-segments have constant length (the road width).
- Cross-segments are perpendicular to both road edges, i.e. locally perpendicular to the centerline of the road.

Saying that cross-segments are normal to both road edges means that they are normal to the tangents to the edges at their end points. Note that this does not generally mean that the tangents to opposite points are parallel. In [4], however, we show that assuming the tangents to opposite points to be approximately parallel is a reasonable assumption in most configurations. This assumption considerably simplifies the recovery of opposite points from the image. It is added to our road model and used in the following section.

3 Conditions for two image points to be matching points

Consider a world road defined by two 3D curves E_1 and E_2, and the road image defined by two image curves e_1 and e_2. Assume that two opposite points A_1 and A_2 on road edges E_1 and E_2 have been found. Their images are a_1 and a_2 (Figure 2), and the following properties follow from the world road model:

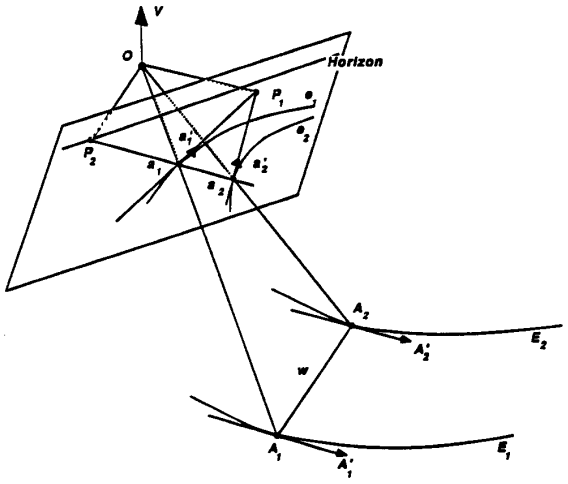

Figure 2: The cross-segment of the world road is assumed horizontal and perpendicular to the tangents at its end points. The tangents are assumed parallel. A condition satisfied by the matching points in the image which also involves the image tangents and the vertical direction is deduced.

1. The segment A_1A_2 is horizontal.
2. The tangents to the road edges at A_1 and A_2 are perpendicular to A_1A_2.
3. The tangents to A_1 and A_2 are approximately parallel.

4. The tangent $\overrightarrow{a_1}'$ to the image edge e_1 at a_1 is the image of the tangent $\overrightarrow{A_1}'$ to the world edge E_1 at A_1; the tangent $\overrightarrow{a_2}'$ to the image edge e_2 at a_2 is the image of the tangent $\overrightarrow{A_2}'$ to E_2 in A_2. This is a general property of projected curves and tangents.

In deriving the following consequences, we make use of the property that the direction of the intersection of two planes is perpendicular to the normals of each plane, and can be obtained by the cross-product of the two normals.

3.1 Directions of tangents to opposite points

If a_1 and a_2 are matching points and $\overrightarrow{a_1}'$ and $\overrightarrow{a_2}'$ are the tangents to the image edges at these points, the direction of the corresponding world tangents is

$$(\overrightarrow{a_1} \times \overrightarrow{a_1}') \times (\overrightarrow{a_2} \times \overrightarrow{a_2}')$$

Proof: If a_1 and a_2 are images of opposite points, the world tangents to the world edges are parallel. Since the images of the world tangents are $\overrightarrow{a_1}'$ and $\overrightarrow{a_2}'$, the world tangents lie on the planes $(\overrightarrow{Oa_1}, \overrightarrow{a_1}')$ and $(\overrightarrow{Oa_2}, \overrightarrow{a_2}')$ respectively. These planes are not parallel since they share the point O and they do not coincide. Since the tangents are parallel, they must be parallel to the intersection of these planes. The direction of this intersection is given by the previous expression.

3.2 Direction of a cross-segment

If a_1 and a_2 are matching points and $\overrightarrow{V}$ the vertical vector, the direction of the world cross-segment is $\overrightarrow{V} \times (\overrightarrow{a_1} \times \overrightarrow{a_2})$.

Proof: A_1A_2 belongs to a horizontal plane since it is horizontal. Since a_1a_2 is the image of A_1A_2, A_1A_2 also belongs to the plane (Oa_1, Oa_2). This plane is generally not horizontal. Thus the direction of the segment A_1A_2 is given by the intersection of a horizontal plane with the plane (Oa_1, Oa_2). The normal to the horizontal plane is the vertical vector $\overrightarrow{V}$. The direction of the normal to the plane (Oa_1, Oa_2) is given by cross-product $(\overrightarrow{a_1} \times \overrightarrow{a_2})$. Thus the direction of A_1A_2 is given by $\overrightarrow{V} \times (\overrightarrow{a_1} \times \overrightarrow{a_2})$.

3.3 Matching condition

If a_1 and a_2 are matching points and $\overrightarrow{a_1}'$ and $\overrightarrow{a_2}'$ are the tangent directions to the image edges in these points, the following relation holds:

$$[\overrightarrow{V} \times (\overrightarrow{a_1} \times \overrightarrow{a_2})] \cdot [(\overrightarrow{a_1} \times \overrightarrow{a_1}') \times (\overrightarrow{a_2} \times \overrightarrow{a_2}')] = 0 \qquad (2)$$

Proof: If a_1 and a_2 are images of opposite points, the direction of the cross-segment A_1A_2 is perpendicular to the direction of the parallel tangents.

3.4 Local normal to the road

If a_1 and a_2 are matching points and $\overrightarrow{a_1}'$ and $\overrightarrow{a_2}'$ are the tangents to the image edges at these points, the local normal to the world road has the direction given by

$$\overrightarrow{N} = [\overrightarrow{V} \times (\overrightarrow{a_1} \times \overrightarrow{a_2})] \times [(\overrightarrow{a_1} \times \overrightarrow{a_1}') \times (\overrightarrow{a_2} \times \overrightarrow{a_2}')] \qquad (3)$$

Proof: The local planar patch of the world road is defined by A_1A_2 and by the parallel tangents at A_1 and A_2. The direction of the normal to this plane is the cross-product of the directions of the cross-segment and of the tangents.

To summarise, when a point a_1 and the tangent $\overrightarrow{a_1}'$ to the road image are given, Equation 2 becomes an equation which must be satisfied by the coordinates of a_2 and the slope of the tangent to the edge in a_2 in order for a_2 to be a matching point to a_1. We can also find the direction of the normal along the corresponding world cross-segment A_1A_2.

4 Search for a matching point of a given image point

If a point a_1 is chosen on one edge image, and if the other edge image is a polygonal line, the matching point a_2 can be located on one of the line segments of the polygonal line, or at one of the vertices between the segments. All the line segments and all the vertices are checked, because a single point a_1 can have several matching point candidates due, for example, to edge irregularities. Other reasons are considered in [4]. For each line segment and for each vertex, the equations developed in the next two subsections are applied.

4.1 Search for a matching point on a line segment

Assume that the segment being considered is the segment p_2q_2. The matching point a_2 is on this segment if

$$\overrightarrow{a_2} = \overrightarrow{p_2} + \lambda \overrightarrow{p_2q_2}$$

with λ between 0 and 1. The point a_2 must also, with its tangent to the edge, satisfy Equation 2. The tangent $\overrightarrow{a_2}'$ to the edge image in a_2 is approximated by the vector $\overrightarrow{p_2q_2}$. We replace $\overrightarrow{a_2}'$, $\overrightarrow{a_2}$ by their values $\overrightarrow{p_2q_2}$, and $\overrightarrow{p_2}+\lambda\overrightarrow{p_2q_2}$ in Equation 2, and transform cross-product combinations into dot products by the well-known identity

$$\overrightarrow{a} \times (\overrightarrow{b} \times \overrightarrow{c}) = (\overrightarrow{a} \cdot \overrightarrow{c})\overrightarrow{b} - (\overrightarrow{a} \cdot \overrightarrow{b})\overrightarrow{c}$$

The resulting value for λ is

$$\lambda = -\frac{(\overrightarrow{V} \cdot \overrightarrow{a_1})(\overrightarrow{K} \cdot \overrightarrow{p_2}) - (\overrightarrow{K} \cdot \overrightarrow{a_1})(\overrightarrow{V} \cdot \overrightarrow{p_2})}{(\overrightarrow{V} \cdot \overrightarrow{a_1})(\overrightarrow{K} \cdot \overrightarrow{p_2q_2}) - (\overrightarrow{K} \cdot \overrightarrow{a_1})(\overrightarrow{V} \cdot \overrightarrow{p_2q_2})) \qquad (4)$$

where $\overrightarrow{K} = (\overrightarrow{a_1} \times \overrightarrow{a_1}') \times (\overrightarrow{p_2} \times \overrightarrow{q_2})$. If λ is between 0 and 1, the intersection is between the end points of line segment p_2q_2, and the value of λ specifies the position of a_2 on p_2q_2. The search also takes place among the vertices between the line segments.

4.2 Search for a matching point at a vertex

We can think of a point q_2 linking two line segments p_2q_2 and q_2r_2 as a point at which the slope of the tangent to the edge changes from the slope of the segment p_2q_2 to the slope of the segment q_2r_2. An approach similar to the previous subsection is followed. A matching point a_2 is at the vertex q_2 if

$$\overrightarrow{a_2}' = \overrightarrow{p_2q_2} + \mu(\overrightarrow{q_2r_2} - \overrightarrow{p_2q_2})$$

with μ between 0 and 1. For this point to be a matching point to a_1, it must also satisfy Equation 2. This produces the following value for μ

$$\mu = -\frac{(\overrightarrow{M} \cdot \overrightarrow{q_2})(\overrightarrow{n_1} \cdot \overrightarrow{p_2r_2}) - (\overrightarrow{n_1} \cdot \overrightarrow{q_2})(\overrightarrow{M} \cdot \overrightarrow{p_2r_2})}{(\overrightarrow{M} \cdot \overrightarrow{q_2})(\overrightarrow{n_1} \cdot (\overrightarrow{q_2r_2} - \overrightarrow{p_2q_2})) - (\overrightarrow{n_1} \cdot \overrightarrow{q_2})(\overrightarrow{M} \cdot \overrightarrow{q_2r_2} - \overrightarrow{p_2q_2}))} \qquad (5)$$

where $\overrightarrow{n_1} = \overrightarrow{a_1} \times \overrightarrow{a_1}'$, $\overrightarrow{M} = \overrightarrow{V} \times (\overrightarrow{a_1} \times \overrightarrow{q_2})$

If the resulting value of μ is between 0 and 1, a matching point a_2 to the point a_1 is located at the vertex q_2.

5 Global road optimization

One of the edge images provides successive points a_1, and for each point, an exhaustive search for matching points a_2 is done for both the line segments of the other edge image and the vertices between the segments, producing several matching candidates a_2. We choose to represent both edge images with a list of linked line segments. The

points a_1 are taken at the midpoints of the line segments, and the tangents $\overrightarrow{a_1}'$ are in the direction of the line segments themselves. For each point a_1, generally one to three points a_2 are found, sometimes more for very noisy images; but generally one point a_2 is a more "correct" matching point for the point a_1 in terms of compatibility with the global road reconstruction. It is also possible that because of noise in the image, the correct match is not among the results, in which case all points a_2 are bad matching points. Thus a method for choosing matching points compatible with a realistic world road is required, and is now discussed.

When a point is chosen on one road image edge, the exhaustive search matches this point with several points on the other road image edge. This *group* of matching points pairs is the image of a group of world cross-segments, but the world road can pass through at most one of these cross-segments. If a sequence of points along one road edge is taken, a sequence of groups of cross-segments is obtained, and the world road must go through at most one of the cross-segments of each group, in the same order as the sequence of points chosen on the first road image edge. Each cross-segment can be represented by a node of a graph (Figure 3). The graph is made up of groups of nodes and a path must be found which visits each group in the proper sequence and goes through at most one node of each group. This path must also maximise an evaluation function which characterises the "goodness" of the road. The total evaluation function is the sum of the evaluation functions of each of the arcs of the graph. The evaluation function for an arc is the sum of weighted criteria C_i which are chosen to characterise a good pairing of two neighbor cross-segments A_1A_2 and B_1B_2. The following criteria were chosen

- The local normal $\vec{N}$ for the cross-segment (Equation 3) should be close to vertical;
  $C_1 = \vec{N} \cdot \vec{V}$ should be close to 1.
- The slope of the patch of two successive cross-segments should be close to vertical.
  $C_2 = [\alpha(\overrightarrow{A_1B_2} \times \overrightarrow{A_2B_1})] \cdot \vec{V}$ should be close to 1 (α is the constant which normalises the vector which follows).
- The average of the directions of the two cross-segments should be perpendicular to the line joining their midpoints (trapezoid constraint, see [3]).
  $C_3 = 1 - \alpha(\overrightarrow{A_1A_2} + \overrightarrow{B_1B_2}) \cdot \alpha(\overrightarrow{A_1B_1} + \overrightarrow{A_2B_2})$ should be close to 1.

Furthermore, when the value of one these criteria falls under an acceptable value (say, cosine of 15 degrees for the road slope criterion C_1), the arc is labelled unacceptable, and cannot be included in a path.

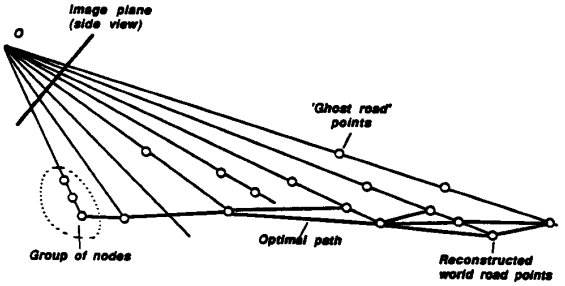

Figure 3: Reconstruction of the world road by a dynamic programming method.

It would also be desirable to introduce constraints such as a requirement for small differences of slope between successive patches, but this type of relation involves three successive cross-segments and complicates the interaction graph. The proposed unary and binary criteria seem to be sufficient for discarding unwanted nodes.

Dynamic programming is an appropriate technique for this type of path optimisation [1]. The following rules were added to the classic dynamic programming technique.

- A node of group $k+1$ may have no acceptable arc extending the paths which reached group k. An arc is labelled unacceptable when any of the criteria which make up the arc evaluation function is under a given threshold. This unconnected node is marked as unusable to the nodes of the next group.
- If in group $k+1$ no node can be linked to any of the nodes of group k, group $k+1$ is discarded altogether and the group of nodes $k+2$ is considered instead, and so on, until a group is found to extend the paths which reached group k. If no further group succeeds in extending the paths, the paths terminate at group k, and among them the path with the largest evaluation function is chosen.

6 Experiments

The algorithm was tested on synthetic road images from roads with variations in bank and width, and on a large number of road images from the DARPA Autonomous Land Vehicle. The results for the synthetic road are presented, and the results from the real world data are discussed.

6.1 Synthetic road image

The road profile from which synthetic images were created is shown in Figure 4. The road centerline profile (side view) is an element of a sinusoid from a crest to a trough (slope downward) or from a trough to

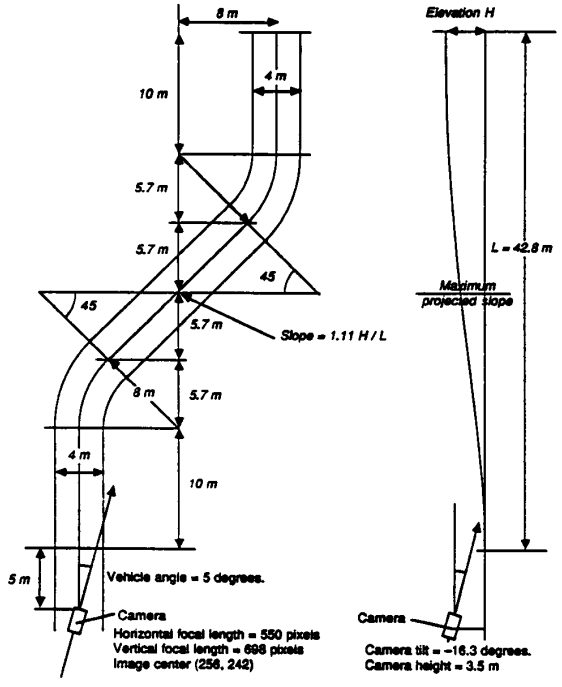

Figure 4: Synthetic road geometry with two 45 degree turns and camera position. Left: Top view. Right: Side view showing the 1/4 period sinusoidal profile of the centerline over the length L.

a crest (slope upward), and the road slope can be modified by varying the sinusoid amplitude. What we call road slope in the following is the slope at the midpoint of the straight piece between the two turns. In this synthetic road it is found equal to $H/38.9$, where H is the difference of level in meters between the lowest and highest point of the road. In top view and going away from the camera, the road has a short straight stretch, then takes a 45 degree left turn and a similar right turn, separated by a short straight line. The camera position, orientation and parameters, also listed in that figure, were taken equal to the values which describe the camera of the ALV.

A benchmark was developed for measuring the performance of the proposed matching points algorithm and other algorithms. A reconstructed road is labelled "usable" if the *centerline* of the reconstructed road stays between the edges of the actual road and does not cut these edges. In other words, a usable reconstruction is a reconstruction in which no cross-segment is off the actual road by more than half its length. When a large number of roads with random variations are considered, percentages of usable roads are calculated.

To obtain the benchmark values described above, random variations are introduced around the nominal values of the road width (4 m) and the road bank (0 degrees). The random width and bank variations are given Gaussian distributions of predefined standard deviations, and several standard deviations are considered for different road slopes. Five road slopes were chosen: −10%, −5%, 0%, 5%, 10%. To reduce the number of combinations of width and bank variations, only

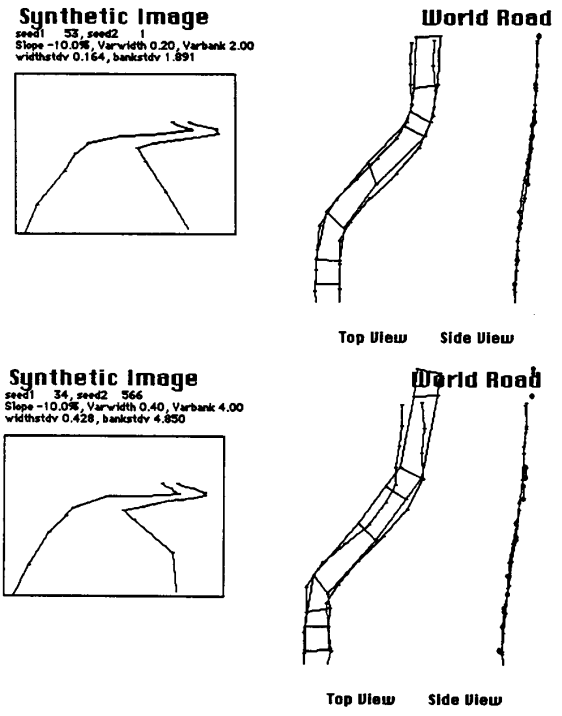

Figure 5: Examples of reconstructions from matching points in images of roads with large random variations in width and bank. Slope and variations are indicated above the images.

the following five width and bank standard deviations were studied for each of the road slopes:

(0 m, 0 degrees), (±0.1 m, ±1 degree), (±0.2 m,±2 degrees), (±0.3 m, ±3 degrees), (±0.4 m, ±4 degrees).

Forty roads were produced for each of these 25 combinations of slopes and standard deviations, and the results for these forty roads were averaged to yield the points shown in the following graphs.

Figure 5 shows two examples of reconstructions for images of downward slopes with large variations of width and bank. Notice that the maximum considered width and bank variations (±0.4 m, ± 4 degrees, bottom left) correspond to very large distortions in the images. The usable length is almost 100% for both examples.

In Figure 6, three algorithms are compared: the present matching points algorithm, the step-by-step (incremental) zero-bank algorithm [2,3], and the Flat-Earth algorithm. A global measure of performance is obtained by averaging the results obtained for the 5 slope configurations, as if the results were averaged from tests over a terrain comprising equal proportions of S-turns at slopes −10%, −5%, 0%, 5%, and 10%. The Flat-Earth algorithm gives reconstructed roads which are 100% usable if the slope is zero, and unusable all the way down if the slope is not zero, independent of width and bank variations; thus the average proportion of usable road produced by the Flat-Earth algorithm is 20%. The averaging over various slopes is more natural for the other two algorithms, because the results are found to be almost independent of the slope.

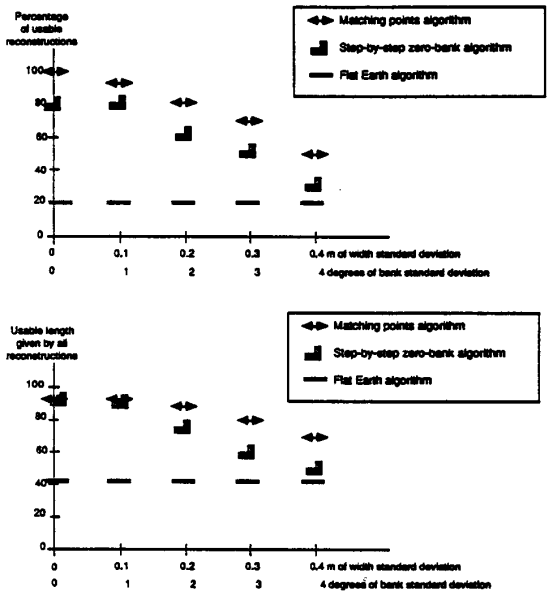

Figure 6: Comparison of the matching points algorithm with the step-by-step zero-bank algorithm and the Flat Earth algorithm. Results for different road elevations were averaged. In this figure, a usable reconstructed road is such that its centerline will not cross the edges of the actual road. Top: Percentage of usable reconstructions for various combined width and bank variations. Bottom: Length of usable reconstructed road length.

Figure 6 shows that the matching point algorithm gives better results than the other two algorithms in term of usable reconstruction. More details can be found in [4].

6.2 Real Imaging

Experiments were also performed with actual road images obtained with the ALV when it was operational at Martin Marietta, Denver. These experiments are described in [9]. Reconstructions were produced for about 50 road configurations including combinations of turns and slope changes. The "ground truth" was provided by a fusion algo-

rithm combining range data and video data; however, the ERIM laser ranger has a limited range of action. Only the first 15 meters of the road could be reconstructed by the fusion method. The reconstruction by the zero-bank algorithm extended at least twice as far in most road configurations. In the short stretch where both reconstructions were available, the agreement was considered good in top view. Differences of elevations appeared in some experiments in side view between the reconstructions of the range-video fusion and the present algorithm, although the difference would probably not have resulted in different steerings of the vehicle. This seems to be due to the fact that we obtained the road width from the Flat-Earth approximation applied to the closest road segment, a method which is sensitive to local bumps under the wheels of the vehicle. In such situations of course the range-video fusion algorithm still produces a correct road profile. Further discussions can be found in [9].

7 Conclusions

In this work, we have derived an analytical condition for points taken on the image of road edges to be matching points, i.e. images of opposite edge points. Taking one point on one road image edge, we generally find more than one candidate matching point. All candidates were back-projected to 3D, and a dynamic programming optimization built a physically acceptable road through the appropriate cross-segments.

A benchmark was applied to compare this algorithm with two others, the step-by-step zero-bank algorithm and the Flat-Earth algorithm. The proposed matching points algorithm was found definitely more effective than the other algorithms over all the considered variations in width and bank.

Experiments with real world data and comparisons with road reconstructions obtained by fusion between video data and range data showed a good agreement in the short range for which range data are available, provided the scaling factor left undefined by the algorithm is well chosen. The proposed algorithm can use the information provided by the range data to calculate this scaling factor, and extend the road reconstruction to the road parts which are in the video camera field of view but out of the range scanner field.

Acknowledgements

The authors would like to thank Peter Meer, Behrooz Kamgar-Parsi and Azriel Rosenfeld for helpful comments and discussions. The support of the Defense Advanced Research Projects Agency and the U.S. Army Engineer Topographic Laboratories under Contract 88-C-0008 (DARPA Order 6350) is gratefully acknowledged.

References

[1] D.H. Ballard and C.M. Brown, "Computer Vision", Prentice-Hall, Inc.

[2] D. DeMenthon, "Inverse Perspective of a Road from a Single Image", Center for Automation Research Technical Report CAR-TR-210, University of Maryland, July 1986.

[3] D. DeMenthon, "A Zero-Bank Algorithm for Inverse Perspective of a Road from a Single Image", in Proc. IEEE Int. Conf. Robotics and Automation, Raleigh, NC, April 1987, pp. 1444–1449.

[4] D. DeMenthon, "Reconstruction of a Road by Matching Edge Points in the Road Image", Center for Automation Research Technical Report CAR-TR-368, University of Maryland, June 1988.

[5] E.D. Dickmanns and V. Graefe, "Dynamic Monocular Machine Vision" and "Applications of Dynamic Monocular Machine Vision", *Machine Vision and Applications*, International Journal, Springer-Verlag International, 1988.

[6] K. Kanatani and D. DeMenthon, "Reconstruction of 3D Road Shape from Images: A Computational Challenge", Center for Automation Research Technical Report CAR-TR-413, December 1988.

[7] K. Kanatani, K. Watanabe and C. Koyama, "Reconstruction of a 3-D Road Geometry from Images for Autonomous Land Vehicles", em IEEE Trans. Robotics and Automation, **RA-5**, no. 13, 1990.

[8] K. Kanatani and K. Watanabe, "Computing 3D Road Shape from Images for ALV: A Challenge to an Ill-Posed Problem", IEEE/RSJ. Int. Workshop on Intelligent Robots and Systems, Tsukuba, Japan, September 1989.

[9] D. Morgenthaler, S. Hennessy, and D. DeMenthon, "Range-Video Fusion and Comparison of Inverse Perspective Algorithms", to be published in *IEEE Trans. on Systems, Man, and Cybernetics.*, Special Issue on Unmanned Systems and Vehicles, Summer 1990.

[10] S. Ozawa and A. Rosenfeld, "Synthesis of a Road Image as seen from a Vehicle", *Pattern Recognition* **19**, 1986, pp. 123–145.

[11] C. Thorpe, M.H. Hebert, T. Kanade, and S.A. Shafer, "Vision and Navigation for the Carnegie-Mellon Navlab", *IEEE Trans. Pattern Anal. Machine Intell.* **PAMI-10**, no. 3, May 1988, pp. 362–373.

[12] M.A. Turk, D.G. Morgenthaler, K.D. Gremban, and M. Marra, "VITS: A Vision System for Autonomous Land Vehicle navigation", *IEEE Trans. Pattern Anal. Machine Intell.* **PAMI-10**, no. 3, May 1988, pp. 342–361.

[13] A.M. Waxman, J. LeMoigne, L.S. Davis, B. Srinivasan, T.R. Kushner, E. Liang, and T. Siddalingaiah, "A Visual Navigation System for Autonomous Land Vehicles", *IEEE J. Robotics and Automation* **RA-3**, no. 2, April 1987, pp. 124–141.