

Patchlets: representing stereo vision data with surface elements

Don Murray* and James J. Little

Department of Computer Science
University of British Columbia
Vancouver, BC, Canada

Abstract

This paper describes a class of augmented surface elements which we call patchlets. Patchlets are planar surface elements generated from dense stereo vision 3D range images. Patchlets have a position, surface normal and size. In addition they have confidence measures on the position and normal direction that are based on the sensor accuracy. These confidence measures facilitate their use with probabilistic methods such as clustering for range image segmentation. Patchlets are formed by the projection of a pixel within the stereo image onto a sensed surface. They are surface elements that are constructed directly from the sensor data and can be used as a fundamental sensed-data primitive.

We describe patchlet formation from the stereo disparity image, the propagation of errors from the stereo sensor model, and confirm experimentally the patchlet model representation. We provide surface segmentation as a sample patchlet application.

1. Introduction

In this paper we investigate using a class of surface elements that we call *patchlets* as the fundamental sensor primitive for stereo vision. Patchlets are a point-based stereo vision sensor primitive which is directly suitable for point-based rendering. Since stereo vision is based on correlation between image regions, it matches surface patches that are seen in more than one camera. Patchlets augment the sensed 3D data point with surface patch properties obtained from the correlation of image regions. Patchlets provide a natural representation of geometric data obtained by stereo vision. Patchlets could also be easily extended to cover triangulation-based laser range scanner.

Stereo vision is a range sensing technology that is inexpensive and generates full field-of-view 3D data in real-time, but its noise characteristics limits its utility for shape scanning. By correctly modeling the stereo sensor noise and

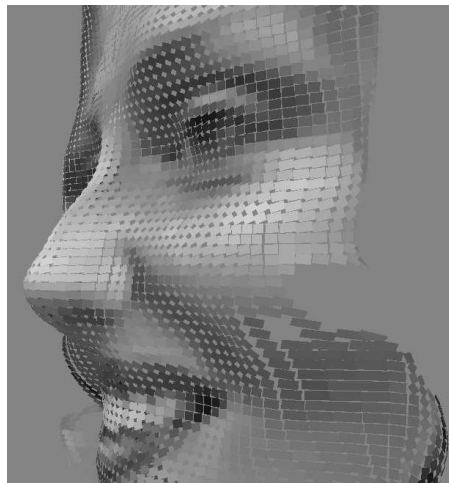


Figure 1: A close-up of patchlets made from a single stereo image - patchlets are drawn at 90% of full size so that their shape can be seen.

propagating this error to the patchlets we obtain meaningful confidence measures on the patchlet parameters. Our contribution is a method to interpret stereo vision with a surface element data primitive that also contains uncertainty measures on the elements position and surface normal. The patchlets can be used directly for rendering, but they can also be used for further shape analysis. Since the patchlets contain accurate confidence measures, they are amenable to probabilistic methods. We demonstrate this with a planar surface segmentation application based on clustering. Figure 1 shows a close-up view of a patchlet cloud generated from the image given in Figure 2. Each small rectangles is one patchlet that maps to one pixel in the stereo image.

In the following section we briefly overview related research. We follow this in Section 3 with a review of stereo vision and the error model which we use. Section 4 gives a detailed description of the creation of patchlets from sensor data and the propagation of errors from disparity space, through 3D points, to the patchlet parameters. In Section 5

*This work was supported by grants from the Natural Sciences and Engineering and Research Council of Canada and from the GEOIDE Network of Centres of Excellence.

we present some example patchlet clouds and give the results of regression testing which confirm the accuracy of the stereo and patchlet models. We describe the segmentation of surfaces from stereo range images as an example application that demonstrates the utility of the patchlet elements in Section 6. Finally, in Section 7 we discuss the possibilities for future extensions of the presented methods.

2. Related work

Use of points for rendering and shape representation is not new but is gaining popularity. Levoy and Whitted [LW85] introduced the idea of points as fundamental rendering primitives. Oriented particles, point/surface-normal pairs, were proposed for particle based modeling by Szeliski and Tonnesen in [ST92], although these particles were developed for shape representation and manipulation rather than rendering. Since 2000 the popularity of particle- or point-based methods has experienced a surge of interest. Inspired by Grossman and Dally’s work on point-based rendering [GD98], Pfister et al. developed *surfels* – oriented particles with area for efficient rendering of complex objects [PZvBG00].

Use of oriented particles for analysis of stereo vision began with Pascal Fua. In [Fua96], he combined stereo vision data from multiple viewpoints into dense point clouds from which he generated oriented particles similar to surfels from the resultant 3D data. He achieved this by organizing the point clouds into voxels and fitting local best-fit surfaces to the data in each voxel that was sufficiently populated. In work inspired by Fua’s particle work, Sara and Bajcsy [SB98] developed their *fish scales* paradigm that is also based on interpretation of stereo data. In their method, rather than fitting planes to the voxelated point data, they create a *fuzzy set* that includes a position and a covariance matrix of the set distribution. By considering the covariance, they can classify the fuzzy set into a class of shape, such as a ball, a plate, or a line. Similar to Fua’s work, their methods are applied to point clouds made from data combined from multiple viewpoints. Carceroni and Kutulakos extract and track structure from multi-view camera temporal sequences using Surfel Sampling [CK].

Patchlets differ from the previous work in a few important ways. Patchlets are inspired by the observation that correlation stereo works by recognizing local surface patches. Therefore, the patchlet element models stereo specific surfaces rather than arbitrary point clouds. For this reason, patchlets are created on an image-by-image basis rather than using point data combined from multiple points of view. As well, we propagate the uncertainty from the stereo-based 3D points to uncertainty measures of the orientation and position parameters of the patchlets. This facilitates the use of probabilistic methods when interpreting

patchlets; they can be properly weighted based on their parameter confidence. Finally, patchlets have a size parameter that is based on the projection of the disparity pixel onto the sensed surface, rather than uniform division of space.

3. Stereo vision

Stereo vision is the process of extracting 3D information from cameras that are physically offset. By identifying pixel locations in two cameras that are known to correspond to the same physical 3D position, the position can be extracted via triangulation. The distance in image space between a feature identified in one camera and its corresponding position in the other camera is called *disparity*. There is an one-to-one (inverse) mapping between disparity and physical distance from the stereo rig.



(a) Reference image



(b) Disparity image

Figure 2: Stereo scene and disparity image

The advantages of stereo vision for range sensing are: it is relatively cheap, fast, and produces a full field-of-view of sample points simultaneously. As well, the points are registered with the appearance image information. It also has many drawbacks. Because of the inverse relationship between disparity and distance, the same error in correspondence causes errors in 3D position that grow dramatically with distance from the camera. The dependence on image neighbourhoods of support causes smoothing in 3D information and has difficulty with thin objects such as railings.

Finally, occlusions, viewpoint limitations, and poor image information can make stereo depth extraction incorrect and incomplete.

Figure 2 shows an example stereo image. (The stereo camera used was a Point Grey Research Digiclops camera. See <http://www.ptgrey.com>.) Figure 2(a) shows the reference camera image and Figure 2(b) shows the dense disparity image generated with sum-of-absolute differences correlation stereo. As disparity is inversely related to distance, in this image the brighter a pixel is, the closer it is to the camera. Black indicates regions of the stereo image for which no disparity value could be determined.

3.1. Stereo error model

There are two classes of errors in stereo disparity images: mismatch errors and estimation errors. Mismatch errors are errors in stereo correspondence and lead to 3D points that are uncorrelated with the true range value. These pixels can be removed from the disparity image using filtering techniques such as given by Fua [Fua93].

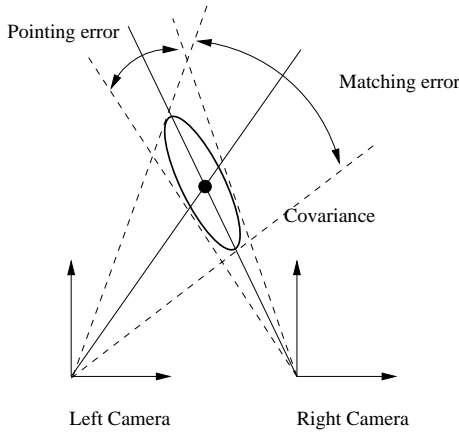


Figure 3: Stereo error model

Estimation errors are slight miscalculations in the disparity value due to local disturbances such as image noise. Matthies and Grandjean showed that these errors can be modeled successfully with a Gaussian distribution [MG94]. We used the estimation error model illustrated in Figure 3. Each disparity pixel from the stereo rig can be converted to a 3D point based on the projective camera equations:

$$\frac{u}{f} = \frac{x}{z}; \quad \frac{v}{f} = \frac{y}{z}; \quad \frac{z}{f} = \frac{B}{d} \quad (1)$$

where (u, v) is the position of the disparity pixel in the reference camera image plane, (x, y, z) is the position of the observed 3D point in the reference camera coordinate frame and d is the pixel disparity. We define our sensor model error to be the combination of two parts: *pointing error* (p)

and *matching error* (m). Pointing error is the variance in (u, v) of the reference camera and is determined by the accuracy of the camera calibration. Matching error is the variance of the disparity d and is determined by the accuracy of the correlation matching. The covariance matrix of the disparity pixel in (u, v, d) space can then be written as:

$$\Lambda_u = \begin{bmatrix} p & 0 & 0 \\ 0 & p & 0 \\ 0 & 0 & m \end{bmatrix} \quad (2)$$

To obtain the covariance matrix (Λ_x) of the 3D point (x, y, z) associated with a disparity pixel (u, v, d) , we propagate this error from (u, v, d) space to (x, y, z) space by applying the methods given in Faugeras [Fau93]:

$$\Lambda_x = J\Lambda_u J^T \quad (3)$$

where

$$J = \begin{bmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} & \frac{\partial x}{\partial d} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} & \frac{\partial y}{\partial d} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} & \frac{\partial z}{\partial d} \end{bmatrix} = \begin{bmatrix} \frac{B}{d} & 0 & \frac{-uB}{d^2} \\ 0 & \frac{B}{d} & \frac{-vB}{d^2} \\ 0 & 0 & \frac{-fB}{d^2} \end{bmatrix} \quad (4)$$

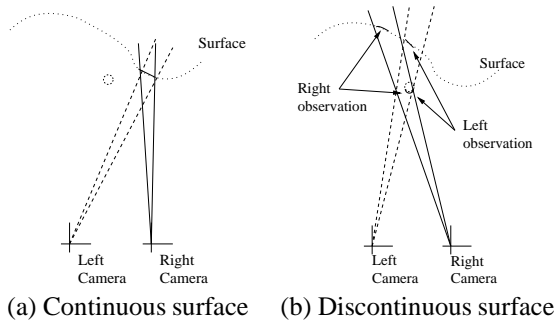
With the above error model, given the pointing and matching error for a stereo camera system we can determine the covariance matrix for each 3D point in the disparity image. This covariance matrix defines the confidence we have in the accuracy of the 3D point position estimation. Patchlet confidence measures are determined by propagating the accuracies of the 3D points on which the patchlet is based.

4. Patchlets

The primary contribution of this paper is the development of the patchlet surface element as a fundamental sensor primitive for stereo vision data. The idea of the patchlet arose from the observation that correlation stereo vision is a region matching technique and consequently senses surface patches, not points. The surface patch corresponds to the portion of the scene that falls within the neighbourhood of a given stereo pixel, as defined by the stereo matching algorithm, (typically a square image mask of 5×5 pixels.)

The justification of this observation is given in Figure 4. As shown in Figure 4(a), when a single continuous surface covers the image neighbourhood, there is a high correlation of the image regions between the stereo cameras. Conversely, as shown in Figure 4(b), if there is a surface discontinuity within the image neighbourhood, there is little correlation between the images and consequently we can expect the stereo algorithm to fail.

Since correlation stereo senses surface patches, a surface element representation for the sensor data is appropriate. One patchlet is generated for each valid pixel in the stereo image. The size of the patchlet is determined by projecting



(a) Continuous surface (b) Discontinuous surface

Figure 4: Surface continuity and correlation stereo

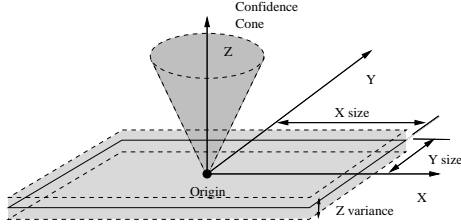


Figure 5: The Patchlet model

the pixel onto the sensed surface. This projection distorts the patchlets so that they form a complete coverage over the imaged surface, however the patchlets are not uniform in size and shape. Patchlets that are farther from the camera will be larger and have higher uncertainty measures than close patchlets. Patchlets on oblique surfaces will be long and narrow rectangles, while patchlets on frontal-parallel surfaces will be square.

Figure 5 illustrates the patchlet model. Each patchlet has its own coordinate system that defines its normal direction, its position in the world coordinate system, and the primary axes of its size parameters. The grey regions in the figure show the regions of uncertainty associated with the position and orientation parameters. Symbolically a patchlet is represented by the letter P and its parameters are:

- X_P the patchlet 3D position in the world coordinate frame;
- ψ_P the patchlet normal vector;
- T_w^P the 4×4 homogenous transform that defines the patchlet's local coordinate system;
- H_P patchlet height (size in local Y);
- W_P patchlet width (size in local X);
- Λ_P the positional variance in normal direction;
- κ_P the confidence in the normal direction.

The positional uncertainty is modeled using a Gaussian distribution with variance Λ_P . The normal direction space is a two-dimensional sphere that wraps around on itself. Consequently, the Gaussian model is unsuitable. Instead we

selected the Fisher [Fis53] distribution which is specially tailored for circular or spherical spaces. The Fisher distribution probability density function takes the form:

$$P(x|\mu, \kappa) = \frac{\kappa e^{\kappa\psi}}{4\pi \sinh(\kappa)} \quad (5)$$

where x is a given vector, μ is the distribution mean vector and κ is the Fisher confidence measure. ψ is the angular difference between μ and x . The Fisher confidence measure κ can be considered as the inverse of the standard deviation in Gaussian distribution.

4.1. Patchlet creation

To extract a patchlet at a particular location in the disparity image we use the neighbourhood around that location in the disparity image to estimate a local planar fit. We chose a square neighbourhood of the equal or smaller size as the stereo correlation mask (for experiments in this paper, we used a support region of 5×5 pixels). The steps of the patchlet creation are:

1. Determine 3D points from the disparity pixels
2. Find the best-fit plane.
3. Find the patchlet origin, coordinate frame and size.
4. Propagate confidence measures.

These steps are elaborated below.

4.1.1 Determine 3D points

We generate the 3D points, $\mathbf{X} = \{X_1 \dots X_n\}$, associated with the disparity pixels within the neighbourhood using (1). Each 3D point also has an independent covariance matrix, Λ_i , determined by (2), (3) and (4).

4.1.2 Find best-fit plane

We find the best plane in a least-squares sense to \mathbf{X} . Since each point has its own covariance matrix and these matrices are not aligned, we required an iterative solution. We selected Levenberg-Marquardt [PTVF92] as an efficient and robust optimisation approach. The error that is minimised is the Mahalanobis distance,

$$e_i^2 = (X_i - \hat{X}_i)' \Lambda_i^{-1} (X_i - \hat{X}_i) \quad (6)$$

where \hat{X}_i is the point on the current plane estimate that minimises the error e_i . \hat{X}_i is not necessarily the perpendicular projection of X_i onto the plane since the error term includes the covariance matrix Λ_i .

To determine \hat{X}_i we perform a whitening or sphering transform [KVV95] to both X_i and the plane estimate. In the whitened space, \hat{X}_i is the perpendicular projection of X_i

onto the plane. \hat{X}_i is determined in the whitened space and then the reverse transform is applied to determine \hat{X}_i . After the Levenberg-Marquardt minimisation has converged, typically in 3 - 4 iterations, the plane normal direction ψ_P is obtained.

4.1.3 Find the patchlet origin, coordinate frame and size

Each patchlet has its own coordinate system. The patchlet origin represents its centre position. The Z axis is the surface normal. For simplicity, we choose to represent patchlets as simple rectangles, rather than as the quadrilateral that is the projection of a square pixel onto an arbitrary surface. We select the X axis to be the larger primary axis of the rectangle and the Y axis the shorter.

The Z axis, or plane normal, is already determined in the previous step. The patchlet origin, X_P , is the projection of the disparity pixel centre onto the planar surface based on the camera parameters. We determine the X and Y axes by finding the major and minor axes of the ellipse that is the projection of a circular pixel onto the planar surface. This is determined by $Y = Z \times R$ and $X = Y \times Z$ where R is the unit vector associated with the ray that connects the reference camera projective centre with the patchlet origin. After the three coordinate axes and the origin are obtained, the world-to-patchlet transform T_w^P can be generated.

The size in the patchlet Y axis direction, H_P , is determined by the frontal-parallel pixel size which is $H_P = \frac{z}{f}$ where z is the z coordinate value of the patchlet origin in the camera coordinate system and f is the camera focal length. The size in the patchlet X axis direction, W_P , is determined by $W_P = \frac{H_P}{\cos \alpha}$ where α is the angular difference between the normal and the vector R .

4.1.4 Propagate confidence measures

The last step in patchlet creation is the propagation of the uncertainty associated with the underlying 3D points to the patchlet parameter space. This determines the confidence parameters for the normal direction and patchlet position. The covariance matrix for the patchlet parameters can be determined by $\Lambda = (J^T J)^{-1}$ where J is the Jacobian matrix defined by $J_{ij} = \frac{\partial e_i}{\partial h_j}$. e_i is the distance off the plane associated with point i as given in (6) and h_j is the j^{th} parameter of the plane. For our plane representation we use three parameters - yaw, pitch and distance from the origin. We evaluated J numerically by performing small perturbations of h and observing the resultant changes in e . Before evaluating J , we first transform the coordinate system so that the plane has yaw and pitch of 0 and the origin is centred at the pointcloud centroid. This moves the system as far as possible from the singularity of a polar-coordinate repre-

sentation of vectors that appears at pitch = $\pm\pi$, (the ‘‘north’’ and ‘‘south’’ pole of the system). As well, placement of the origin at the centroid of the point set ensures the greatest independence between the angles and position ([Mur04] discusses angle and offset parameter independence).

For our 3-parameter plane, $\{\theta, \phi, o\}$ (where $\theta = \text{yaw}$, $\phi = \text{pitch}$ and $o = \text{positional offset}$) the resulting covariance matrix is 3×3 . The upper-left 2×2 submatrix is the covariance matrix, $\Lambda_{\theta\phi}$, of the angles $\{\theta, \phi\}$ and the lower-right value is Λ_P , the variance of offset parameter o . κ_P is extracted from $\Lambda_{\theta\phi}$ by determining the direction of maximum variance and calculating the corresponding Fisher spherical confidence measure.

5. Patchlet results

Figure 6 shows the patchlet cloud generated from the stereo image given in Figure 6. The patchlets are rendered from two novel viewpoints to illustrate the quality of the depth reconstruction. In the left images, each patchlet is displayed with the greyscale of associated pixel in the reference image. In the right images, each patchlet is coloured white and Gouraud shaded. Each patchlet is the size of the pixel from which it was generated, projected onto the sensed surface. The patchlets are drawn at 90% of their full size so that the gaps between patchlets are easily visible.

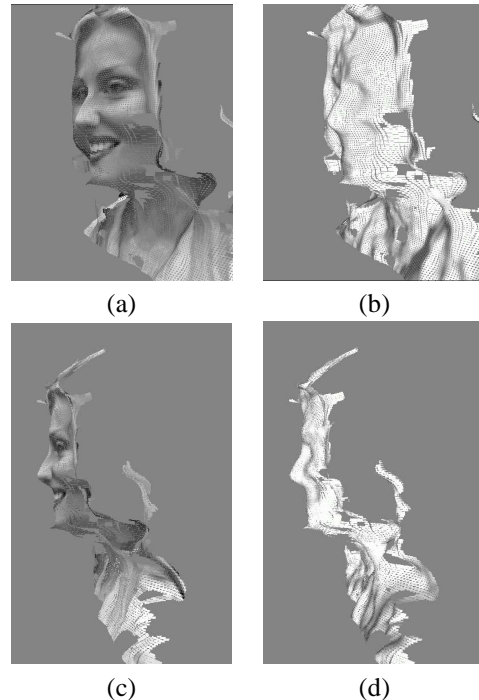
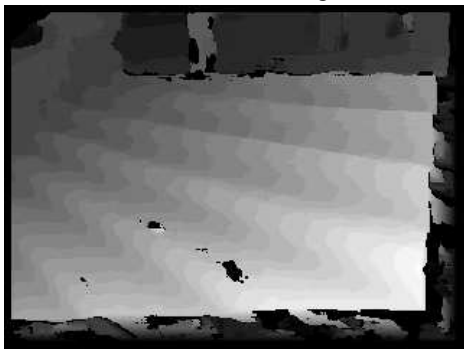


Figure 6: Patchlets rendered from two views : (a) and (c) are colored by the source image, (b) and (d) are white with Gouraud shading

Figure 1 (on first page) shows a close-up view of the foreground subject. In this view the individual patchlets are clearly visible. One can see that patchlets on more oblique surfaces, such as the lower cheek, are considerably larger in order to cover the entire surface continuously. There is some smoothing of features caused by the plane-fitting process of patchlet creation, but most observed smoothing (such as between the lower nose and the upper lip) is actually caused by the correlation stereo and is observable in the underlying point cloud.



(a) Reference image



(b) Disparity image

Figure 7: Stereo scene and disparity image of outdoor steps

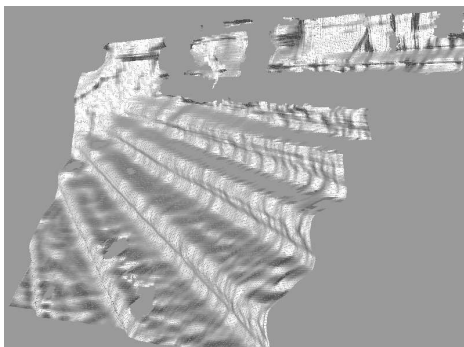
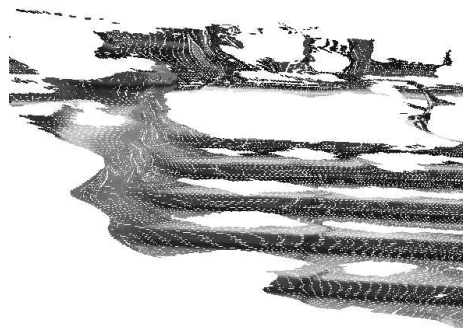
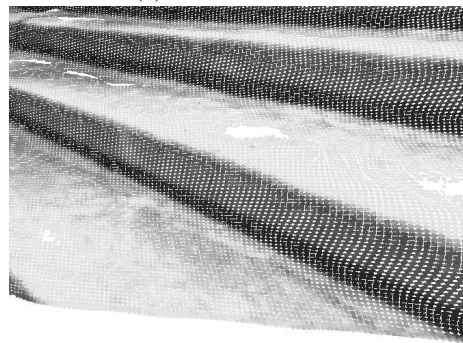


Figure 8: Patchlets generated from scene in Figure 7 and rendered from a novel viewpoint, Gouraud shaded

Figure 7 gives a reference/disparity image pair similar to Figure 2, this time of an outdoor scene. The distances are larger than in Figure 2 and consequently the errors are greater, especially for farther regions of the image such as the staircase balustrade. Figure 8 shows the generated patchlet cloud rendered from a novel viewpoint. One can see in that there is considerable wobble in the represented surface due to stereo estimation errors. Figure 9 shows two additional views of the patchlet cloud. Figure 9(a) shows the balustrade viewed from the front of the steps. Significant smoothing between the steps and the balustrade can be seen. This is caused by the stereo correlation mask during the correlation algorithm.



(a) Far side of stairs



(b) Close up

Figure 9: Scene rendered from additional viewpoints

5.1. Accuracy analysis

The pointing accuracy required by (2) is supplied by the calibration process of the stereo camera. In our case it was supplied as a standard deviation of 0.03 pixels when operating with 320×240 resolution stereo images. However, the matching accuracy, m , is a feature of the correlation algorithm and is less straightforward to estimate. We derived the matching accuracy experimentally.

To determine the matching accuracy we took a stereo image of a well textured planar surface (in this case, carpet

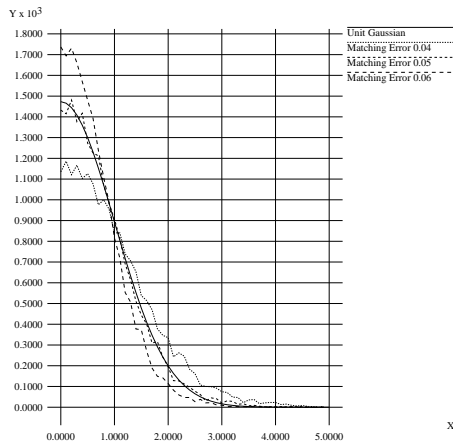


Figure 10: Determination of stereo matching accuracy: displayed is the histogram of Mahalanobis distances for three matching errors overlaid on the histogram expected from the Unit Gaussian. The Y axis is number of points, the X axis is in standard deviation.

over a concrete floor). We then postulated several values for the matching error and evaluated their accuracy. Given the manufacturer-specified pointing accuracy and a hypothesis for the matching accuracy, we calculated the covariance matrix, Λ_x , for each disparity pixel in the stereo image as given by (3), as well as the disparity pixel's x, y, z location. We then generated a normalized histogram of the Mahalanobis distance of each calculated 3D location from the plane. The histogram that best matched the Unit Gaussian would be the one with the accurate value for the matching error, (i.e., we iterated through a set of matching errors, searching for the one in which exactly 67% of the 3D points were within one standard deviation of the plane, 95% were within two standard deviations and so on.)

The results of this search are shown in Figure 10. This test verifies two issues. First, that the histogram shape conforms to a Gaussian, verifying that the assumption of a Gaussian distribution for sensed data points is accurate. Second, it determines that a matching accuracy of 0.05 pixels standard deviation best modeled our system.

As a sanity test, we then performed a regression test on the confidence measures for the patchlets created from the same planar surface scene. We used the pointing and matching errors as determined above, generated patchlets based on the resulting sensor model, and examined the histogram of the angular difference between the patchlet normals and the plane normal, normalised by the orientation confidence measure. The resulting histogram is displayed in Figure 11. The dashed line represents the unit Gaussian, our expected histogram level. The solid line represents the obtained histogram, which matches the Unit Gaussian very closely, al-

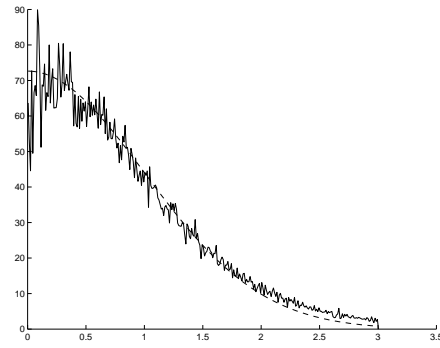


Figure 11: Verification of patchlet accuracy: displayed is the histogram of Mahalanobis distances for normal direction of a patchlet cloud from its generative plane, overlaid with a Unit Gaussian. The Y axis is number of patchlets, the X axis is in standard deviation.

beit with a slightly heavier tail.

6. Range image segmentation

Patchlets were originally conceived as a tool for interpretation of stereo range data for mobile robot navigation. Mobile robots can collect a large amount of data from their environments, and a compact representation of the environment is desirable for mapping and navigation algorithms.

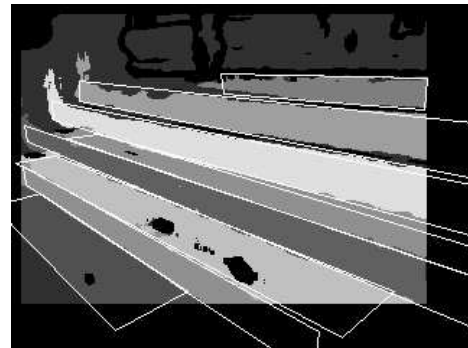


Figure 12: Surface segmentation

Figures 12 and 13 show the results of probabilistic planar surface segmentation from the patchlet cloud from Figure 7 using clustering. The details of the segmentation process are described fully, and more complete results are presented, in [ML]. The method use Expectation-Maximisation to maximise the joint probability of the patchlet data and the segmented surfaces. Figure 12 uses the full patchlet model and includes dimensions for both surface normal and surface connectivity. Figure 13 gives the results of the clustering using only the patchlet position information. We can see from these figures that the added dimensionality of normal direction, surface contiguity and

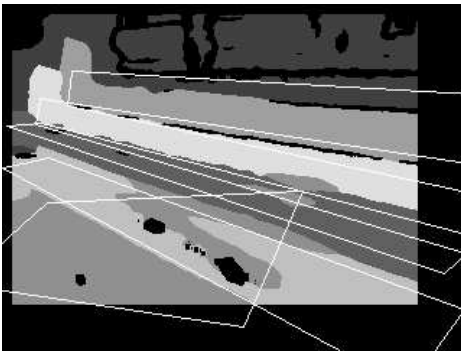


Figure 13: Segmentation using point information only

proper weighting with the uncertainty measures greatly improves the quality of the segmentation.

7. Conclusions

Point-based methods are gaining popularity in the graphics community for their ability to handle complex shapes without constructing or maintaining explicit connectivity between points. Stereo vision is a sensing technology that has considerable promise, it is inexpensive, fast and simple. In our research we are bringing these two areas together, demonstrating that point-based data primitives are a natural and useful way to interpret stereo vision 3D data.

Stereo vision is a noisy sensor, consequently maintaining uncertainty data is a requirement for later shape analysis and especially important if one wishes to merge data from multiple viewpoints. Patchlets, our primitives, may be used directly for modeling as well as for data-driven shape analysis. We have demonstrated this usefulness through application to larger surface extraction and estimation.

7.1. Future work

A stereo image is quite limited in its ability to reconstruct scenes due to its viewpoint limitation. For more complete models it is necessary to register views and combine the resultant 3D data. The uncertainty measures of patchlets make it well suited for the process of combining data. The certainty and the sizes of the patchlets can play an important role in maximising the accuracy of the final result. We are working to extend the application of patchlets to the combination of multi-viewpoint data. As well, the example application in this paper is restricted to extracting planar surfaces. This application was designed for constructing environment models of indoor environments for mobile robots, in which the planar surface constraint is largely accurate. However, for more complex and organic environments or objects, expanding the shape analysis to higher order parameterised surfaces would be advantageous.

References

- [CK] Rodrigo Carceroni and Kiriakos Kutulakos. Multi-View scene capture by surfel sampling: From video streams to Non-Rigid 3D motion, shape & reflectance. In *ICCV 2001*, pages 60–67.
- [Fau93] O D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.
- [Fis53] R.A. Fisher. Dispersion on a sphere. *Proceedings of the Royal Society London*, A127:295–305, 1953.
- [Fua93] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image feature. *Machine Vision and Applications*, 6:35 – 49, 1993.
- [Fua96] P. Fua. Reconstructing complex surfaces from multiple stereo views. In *Proceedings of the Fifth International Conference on Computer Vision (ICCV '96)*, pages 1078 – 1085, Cambridge, Massachusetts, USA, June 1996.
- [GD98] J. Grossman and W. Dally. Point sample rendering. In *Eurographics Rendering Workshop*, 1998.
- [KWW95] J. Karhunen, L. Wang, and R. Vigario. Nonlinear PCA type approaches for source separation and independent component analysis. In *ICNN-95*, Perth, Western Australia, November 1995.
- [LW85] M. Levoy and T. Whitted. The use of points as display primitives. Technical Report TR 85-022, The University of North Carolina at Chapel Hill, Department of Computer Science, 1985.
- [MG94] L. Matthies and P. Grandjean. Stochastic performance modeling and evaluation of obstacle detectability with imaging range sensors. *Robotics and Automation (RA)*, 10:783–792, 1994.
- [ML] D. Murray and J. Little. Surface segmentation from stereo scenes. In *3DPVT 2004*.
- [Mur04] D. Murray. *Patchlets: a method of interpreting correlation stereo 3D data*. PhD thesis, University of British Columbia, Vancouver, Canada, 2004.
- [PTVF92] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.
- [PZvBG00] H. Pfister, M. Zwicker, J. van Baar, and M. Gross. Surfels: Surface elements as rendering primitives. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 335–342. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.
- [SB98] R. Sara and R. Bajcsy. Fish-scales: Representing fuzzy manifolds. In *Proceedings of the 6th IEEE International Conference on Computer Vision (ICCV '98)*, pages 811–817, 1998.
- [ST92] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics*, 26(2):185–194, July 1992.