Robust Outlier Removal from Point Clouds Acquired with Structured Light

J. Köhler¹, T. Nöll¹, G. Reis¹ and D. Stricker¹

¹German Research Center for Artificial Intelligence, Trippstadter Str. 122, D-67663 Kaiserslautern, Germany

Abstract

We present a method for automatic, robust identification of outliers in point clouds acquired with structured light. In contrast to most state of the art methods, we consider available 2D information instead of operating on the points only. Our method performs robust even for complex, glossy surfaces, where illumination artifacts introduce ghost geometry or outliers that seamlessly blend into the correct surface. The key idea of our algorithm is to only use cameras for reconstruction and the projector for a separate consistency check. We use encoding by phase shifting to obtain per pixel correspondences for the cameras and subpixel accuracy for the projector. The method requires at least 2 cameras, a projector and a precise calibration of all devices.

Categories and Subject Descriptors (according to ACM CCS): I.4.1 [Image Processing and Computer Vision]: Digitization and Image Capture—Scanning

1. Introduction

The usability of a reconstructed point cloud is significantly decreased the more outliers it contains. Especially when complex, shiny surfaces are reconstructed, global illumination artifacts and specular lobes can cause ghost geometry and outliers that seamlessly blend into the correct surface or are closely distributed around it (Figure 1, top). In order to identify those outliers, existing state of the art postprocessing methods often operate on the 3D geometry only [SBS05, WPH*04, YCC09]. However, the results are often not satisfying: Outliers close to correct points might be treated as noise and smoothed into the final surface, while ghost geometry is itself consistent and cannot be distinguished from the correct parts.

Instead of operating on the 3D points only, we additionally use 2D information that is provided by the calibration of the scanning devices for filtering. Image based reconstruction methods usually establish dense 2D correspondences among the input images. A point cloud is then reconstructed by computing 3D points that yield small reprojection errors when backprojected to these correspondences [HZ04]. The presence of the aforementioned outliers however shows, that a substantial amount of points yields a small reprojection error but is nevertheless wrong. We address this problem by using a structured light scanning setup, where only cameras are used for computing the 3D geometry. The projector does not participate in the triangulation, but serves as an additional control instance. A vertex reconstructed within the bounds of the camera reprojection error is verified by checking its reprojection error with the projector. On the one hand, this circumvents that the nonlinear gamma transfer function between a camera and a projector introduces errors to the reconstruction. On the other hand, the additional information...
constituted by the projector is not disregarded, but serves as a powerful outlier filter. This makes our algorithm robust also in regions close to the correct surface. Outliers are identified with excellent precision, while the correct parts of the reconstructed point clouds are left almost untouched. The point clouds can thus be used for further processing immediately.

2. Geometry Acquisition

The significance of the reprojection error that a vertex scores when it is projected onto the image plane of a projector strongly depends on the granularity of the projected patterns. We thus use phase shifted structured light, which yields correspondences between a pixel \( p_i \) to another camera \( C_j \) that minimizes Equation 1, as seen by camera \( C_i \), \( \Phi_j(p_j) \) denotes the recovered phase as seen by camera \( C_i \), \( \Phi_j(p_j) \) = \( (y, x)^T \), \( x, y \in [0..1] \) are transmitted from the projector to all cameras by encoding them into sinusoidal fringe patterns that are shifted \( n \) times. In the following, \( \Phi_j(x, y) = (x, y)^T \) and \( \Phi_j(x, y) = (y, x)^T \), \( x, y \in [0..1] \) are transmitted from the projector to all cameras by encoding them into sinusoidal fringe patterns that are shifted \( n \) times. In the following, \( \Phi_i = (\Phi_{ix}, \Phi_{iy}) \) denotes the recovered phase as seen by camera \( C_i \), \( \Phi_i(p) \in [0..1] \times [0..1] \) for a camera pixel \( p \) (Figure 2). Several methods exist for recovering \( \Phi_i \) from \( n \) phase shifts, we use the least squares approach discussed in [GZC07]. This method yields \( \Phi_i \) wrapped to the respective intervals. We unwrap it by applying the temporal phase unwrapping approach proposed in [WNB10].

Correspondences between a pixel \( p_i \) of camera \( C_i \) and a pixel \( p_j \) of camera \( C_j \), \( i \neq j \) can be deduced from \( \Phi_i, \Phi_j \) with the following criterion:

\[
\left| \Phi_i(p_i) - \Phi_j(p_j) \right| < \epsilon \Phi
\]

We reconstruct those parts of the surface that are lit by the projector and seen by multiple cameras using a method similar to [HLZ10]. For each pixel \( p_i \) of each camera \( C_i \) that has a valid phase \( \Phi_i(p_i) \), we compute a vertex \( V \) on the ray \( r_{pi} \) through \( p_i \). \( V \) is initialized by triangulating a pixel correspondence \( p_j \) to another camera \( C_j \) that minimizes Equation 1. The depth of \( V \) on \( r_{pi} \) is then optimized iteratively among all \( m \) cameras where the projection of \( V \) fulfills Equation 1, using the Levenberg Marquardt algorithm. As last step, the phase values \( \Phi_j(p_j) \) of all \( m \) observing cameras are attached to \( V \).

Note, that in contrast to other methods (e.g. [ZH04]), we do not use the projector for triangulation. The major reason for this is the nonlinear gamma transfer function between the projector and a camera, that would be propagated into the geometry within the bounds of the reprojection error. As a result, the fringe oscillations are visible in the reconstruction (Figure 3). A possible solution to this problem would be the calibration of the gamma transfer function [WNB10]. This works, however, only for a single camera-projector combination, because it changes the projected pattern with respect to a single camera. Moreover, we found that a gamma calibration can reduce the spectrum of values that are projected, depending on the response of the projector’s actual gamma function and the sensitivity of the camera used.

3. Filtering

A point cloud reconstructed by the algorithm described in Section 2 is not affected by the nonlinear gamma transfer function of the projector, but can still contain many outliers. Especially when reconstructing glossy surfaces, view dependent illumination artifacts like specular lobes and flares cause phase errors for a camera in regions where other cameras observe the correct values. Those errors can emerge continuously, due to the fading of a lobe. Correspondences among the cameras in such a region are wrong, but often within the bound of Equation 1. Reconstructed points are not necessarily noisy, but can form a consistent, displaced surface patch, that seamlessly blends into the correct surface in the worst case (Figure 4). Another typical, unwanted artifact is ghost geometry caused by indirect illumination that is consistently observed by all cameras. When the projector is used as a filtering instance, such outliers can be identified with excellent precision, while the correct parts of the geometry are left almost untouched.

The formation of outliers is illustrated in Figure 5. The left image shows, how phase values are distributed among all devices when no error occurs. Light projected onto a surface is consistently reflected to the cameras. In the right image, \( C_1 \) captures a wrong phase value \( t \). Consequently, the
pixel correspondence \( p_1 \leftrightarrow p_3 \) with \( C_2 \), that was established using the phase values is wrong. When \( p_3 \) lies close to the epipolar line induced by \( p_1 \), however, the reprojection error is low enough for a successful triangulation of a vertex \( V \). It is obvious from Figure 5 (right), that \( V \) will still score a bad reprojection error with the projector, because its ray to \( V \) does not pass through the correct location of \( t \) on the surface.

As described in Section 2, \( V \) is associated with \( m \) phase values \( \phi_i \), when it is observed by \( m \) cameras. These values vary within the bound of \( \varepsilon_\Phi \) (Equation 1). For each \( \phi_i \), the pixel of the projector that should have illuminated \( V \) is computed as

\[
P_{\phi_i} = (\Phi_{ix}, res_x, \Phi_{iy}, res_y)^T
\]

where \( res_{ix/y} \) is the resolution of the projector. Note that \( p_{\phi_i} \) is computed with subpixel accuracy because the projected, continuous \( (\Phi_{ix}, \Phi_{iy}) \) is known. All \( m \) phase values thus yield \( m \) slightly different pixels for the projector.

We consider \( V \) as invalid, when its projection into the image plane of the projector scores a reprojection error larger than \( \varepsilon_p \) with one of the \( m \) reference pixels:

\[
||P(V) - p_0|| > \varepsilon_p
\]

where \( P(V) \) denotes the projection of \( V \) into the image plane of the projector. The reliability of this filtering criterion with respect to a single camera depends on the included angle \( \alpha_t \) between the projector ray and the ray of those cameras \( C_i \), that observe the correct phase value (Figure 5). For a small \( \alpha_t \), \( V \) will be projected closer to the correct projector pixel, even if the vertex is wrong. This is important, when a small amount of cameras (i.e. 2 or 3) is used, which have different angles towards the projector. In this case \( \varepsilon_p \) should be weighted linearly with \( \alpha_t \), that it gets smaller when \( \alpha_t \) decreases.

\[
\varepsilon_p = \varepsilon_p^t \cdot (1 - \frac{2 \cdot \cos^{-1}(r_{Ci} \cdot r_p) - \frac{\pi}{2}}{\pi})
\]

where \( r_{Ci} \) and \( r_p \) are the normalized camera/projector rays through \( V \) and \( \varepsilon_p^t \) is a fixed reprojection threshold depending on the projector resolution.

### 4. Results

We test our method by applying it to both real and synthetic data. We use an acquisition setup with 7 cameras (Canon® EOS 500D) and 1 projector (Sanyo® Z4000) and project the phase encoded by fringe images with 3 and 10 phase shifts. The real datasets are scans of a bowling ball. It was burnished with car polish to increase the specular reflection component, which ensures a certain amount of errors. The synthetic data sets are scans of a perfect sphere that were simulated by ray tracing. We model global illumination artifacts in the scans by applying the Phong reflection model [Pho75] with 12 additional, static light sources (Figure 6).

We fit a sphere to both the filtered and the original, outlier afflicted datasets using RANSAC [FB81] for outlier rejection. The sphere parameters derived from the best RANSAC consensus set are optimized iteratively by applying the Levenberg Marquardt algorithm. For each fit we then compute
## Table 1: Experimental results for all datasets. Mean and max diff abbreviate the mean and maximum difference to the radius.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>radius</th>
<th>mean diff</th>
<th>max diff</th>
<th>σ</th>
<th>#points</th>
<th>% removed</th>
<th>ϵₚ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synthetic sphere, 3 phase shifts</td>
<td>180</td>
<td>0.01717</td>
<td>0.239</td>
<td>0.1534</td>
<td>19113296</td>
<td>8.2%</td>
<td>0.5</td>
</tr>
<tr>
<td>filtered</td>
<td>180</td>
<td>0.009356</td>
<td>0.01625</td>
<td>17542392</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Synthetic sphere, 10 phase shifts</td>
<td>180</td>
<td>0.009501</td>
<td>0.01557</td>
<td>0.06634</td>
<td>19457187</td>
<td>1.9%</td>
<td>0.5</td>
</tr>
<tr>
<td>filtered</td>
<td>180</td>
<td>0.009238</td>
<td>0.01557</td>
<td>19068114</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bowling ball, 3 phase shifts</td>
<td>99.05</td>
<td>0.09869</td>
<td>7.54</td>
<td>0.166</td>
<td>5605980</td>
<td>18%</td>
<td>0.7</td>
</tr>
<tr>
<td>filtered</td>
<td>99.01</td>
<td>0.08985</td>
<td>0.7928</td>
<td>0.1104</td>
<td>4591426</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bowling ball, 10 phase shifts</td>
<td>99.01</td>
<td>0.08393</td>
<td>4.692</td>
<td>0.1574</td>
<td>6040120</td>
<td>9.5%</td>
<td>0.7</td>
</tr>
<tr>
<td>filtered</td>
<td>99.02</td>
<td>0.06634</td>
<td>0.6649</td>
<td>0.1156</td>
<td>5462529</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Three important conclusions can be drawn from the listed data. First, the amount of filtered vertices is lower for the datasets with 10 phase shifts. The least squares phase estimation tends to yield better phase values $\phi_i$, when more shifts are used. This is also obvious when comparing the standard deviation of the unfiltered and 10 shift datasets with each other: $\sigma$ is distinctly lower, when 10 shifts are used. Second, the filtered datasets yield a significantly lower $\sigma$ than the unfiltered datasets. This indicates that the point cloud filtered with our method is less noisy and has a better quality than the unfiltered point cloud. Third, the standard deviations for the filtered data sets are very similar to each other for different amounts of phase shifts. This holds for both real and synthetic datasets and demonstrates the sustainable quality of the filtering. Even with only 3 phase shifts, the filtered sphere has similar characteristics than the one acquired with 10 shifts.

Two types of values are remarkably high: The peaks of the radius deviation in the unfiltered, synthetic datasets result from single, large outliers. The filtering rate for the bowling ball scanned with 3 shifts is large, because its reconstructed border is noisy. The ball’s surface is very challenging for a diffuse structured light scanner because it has, in addition to its glossiness, a metallic like finish that causes coarse grained noise (Figure 6, left) in the fringe images.

## 5. Conclusion

We have shown that the projector of a structured light scanning setup can be used as a powerful instance for filtering outliers. This is more appropriate than integrating it in the triangulation process, because then the nonlinear gamma function is not propagated into the reconstructed points. The method removes almost all outliers, but leaves the correct parts of the geometry untouched.

## References


