Abstract

We present an algorithm for aligning two colored point clouds. The key idea is to optimize a joint photometric and geometric objective that locks the alignment along both the normal direction and the tangent plane. We extend a photometric objective for aligning RGB-D images to point clouds, by locally parameterizing the point cloud with a virtual camera. Experiments demonstrate that our algorithm is more accurate and more robust than prior point cloud registration algorithms, including those that utilize color information. We use the presented algorithms to enhance a state-of-the-art scene reconstruction system. The precision of the resulting system is demonstrated on real-world scenes with accurate ground-truth models.

1. Introduction

We are concerned with the following problem: given two roughly aligned three-dimensional point clouds, compute a tight alignment between them. This is a well-known problem in computer vision, computer graphics, and robotics. The problem is typically addressed with variants of the ICP algorithm [1, 3, 31]. The algorithm alternates between finding correspondences and optimizing an objective function that minimizes distances between corresponding points. A common failure mode of ICP is instability in the presence of smooth surfaces [14, 46]. The alignment can slip when geometric features do not sufficiently constrain the optimization.

This ambiguity can be alleviated if the points are associated with color. This is often the case. Modern depth cameras commonly produce pairs of depth and color images. Many industrial 3D scanners are also equipped with synchronized color cameras and provide software that associates color information with the 3D scans. Multi-view stereo pipelines reconstruct colored point clouds from image collections [8, 13, 39]. Considering color along with the geometry can increase the accuracy of point cloud registration.

The standard formulation for integrating color into geometric registration algorithms is to lift the alignment into a higher-dimensional space, parameterized by both position and color. Typically, correspondences are established in a four- or six-dimensional space rather than the physical three-dimensional space [21, 22, 27, 28]. This is an elegant approach, but it is liable to introducing erroneous correspondences between points that are distant but have similar color. These correspondences can pull away from the correct solution and prevent the method from establishing a maximally tight alignment.

In this work, we develop a different approach to aligning colored point clouds. Our approach establishes correspondences in the physical three-dimensional space, but defines a joint optimization objective that integrates both geometric and photometric terms. A key challenge is that color is only defined on discrete points in the three-dimensional space. To optimize a continuous joint objective, we need to define a continuous and differentiable photometric term, the gradient of which indicates how color varies as a function of position. This is challenging because unstructured point clouds do not provide a natural parameterization domain. We build on dense and direct formulations for RGB-D image alignment, which use the two-dimensional image plane as the parameterization domain [35, 25, 44, 40]. To define a photometric objective for point cloud alignment, we introduce a virtual image on the tangent plane of every point, which provides a local approximation to the implicit color variation. Using this construct, we generalize the photometric objectives used for RGB-D image alignment to unstructured point cloud alignment. The resulting photometric objective is integrated with a geometric objective defined using the same virtual image planes. This enables efficient joint photometric and geometric optimization for point cloud alignment. Our formulation unifies RGB-D image registration and colored point cloud registration. We show that our algorithm achieves tighter alignment than state-of-the-art registration algorithms, including those that use color information.

Our primary contribution is a new approach to colored point cloud registration. Beyond this, we make two supporting contributions. Since point cloud registration plays a central role in high-fidelity scene reconstruction, we have used the presented algorithms to enhance a state-of-the-art
scene reconstruction system [4]. To quantitatively evaluate
reconstruction accuracy on real-world scenes, we have cre-
ated a dataset of indoor scenes scanned with an industrial
laser scanner. Experiments demonstrate that the enhanced
pipeline produces significantly more accurate reconstruc-
tions.

2. Related Work

The ICP algorithm [1, 3, 31] has been a mainstay of ge-
ometric registration in both research and industry for many
years. Its variants have been extensively studied [31, 33,
38]. Notably, point-to-plane ICP has been broadly adopted
due to its fast convergence [3, 31]. ICP and other local re-
finement algorithms require a rough initial alignment as in-
put. Such initial alignment can be obtained via global reg-
istration algorithms [18, 43, 47]. These global algorithms
address a more difficult problem since they must establish
correspondences with no initialization. While significant
progress in global alignment has been made, the alignment
produced by state-of-the-art global registration algorithms
can often be improved by local refinement.

Most local registration algorithms that utilize color in-
formation lift the problem to a higher-dimensional space,
which is used to establish correspondences [21, 22, 28, 27].
Godin et al. [15] use color to prune correspondences. Our
approach is different in that we establish correspondences in
the physical 3D space inhabited by the point clouds that are
being registered, but optimize a joint photometric and geo-
metric objective. A recent work [7] represents color infor-
mation in a Gaussian mixture model. It is built upon a prob-
abilistic registration algorithm [11] and is orders of magni-
tude slower than common ICP variants or our approach.

Many approaches to RGB-D image registration have
been explored. Huhle et al. [20] and Henry et al. [17] com-
bine image matching with geometric registration. Other
approaches optimize a direct photometric objective defined
densely over the images [35, 25]. Whelan et al. [40] intro-
duce a joint optimization objective that combines the pho-
tometric objective and a point-to-plane ICP objective. We
build on these works, specifically on the dense and direct
formulations for RGB-D image registration. We review the
photometric objective used for RGB-D image registration
in Section 3 and then show that it can be generalized to un-
structured point clouds. A key challenge that distinguishes
point clouds from RGB-D images is the lack of a regular
grid parameterization.

Dense reconstruction from RGB-D sequences has been
extensively studied [29, 17, 24, 10, 4, 41]. Such recon-
struction systems commonly have three key components:
surface alignment (in the form of odometry and loop clo-
sure), global optimization, and surface extraction. We show
that the colored point cloud registration approach presented
in this paper can be used to increase the accuracy of the
surface alignment step in a state-of-the-art reconstruction
pipeline, significantly increasing the accuracy of the final
reconstruction. To evaluate this quantitatively, we collect
a dataset of RGB-D video sequences with dense ground-
truth 3D models acquired with an industrial laser scan-
ner. Many RGB-D datasets have been collected in prior
work [37, 42, 19, 5, 12]. To our knowledge, none of them
are accompanied by dense and accurate ground-truth 3D
models of whole scenes. Synthetic datasets have been cre-
ated for this purpose [16, 4]. We complement these efforts
with real-world datasets.

3. RGB-D Image Alignment

In this section, we review the photometric objective for
RGB-D image alignment [35, 25] and combine it with a ge-
ometric objective defined on the same image plane. This in-
roduces notation and lays the groundwork for colored point
cloud alignment, which will be presented in Section 4.

An RGB-D image is composed of a color image $I$ and a
depth image $D$ registered to the same coordinate frame. For
simplicity we use intensity images. Given a pair of RGB-D
images $(I_i, D_i)$ and $(I_j, D_j)$ and an initial transforma-
tion $T^0$ that roughly aligns $(I_i, D_i)$ to $(I_j, D_j)$, the goal is to
find the optimal transformation that densely aligns the two
RGB-D images.

A photometric objective $E_I$ is formulated in terms of
squared differences of intensities [35, 25]:

$$E_I(T) = \sum_x (I_i(x') - I_j(x))^2,$$

(1)

where $x = (u, v)^T$ is a pixel in $(I_j, D_j)$ and $x' = (u', v')^T$
is the corresponding pixel in $(I_i, D_i)$. The correspondence
is built by converting the depth pixel $(x, D_j(x))$ to a 3D
point in the camera space of $(I_j, D_j)$, transforming it with $T$,
and projecting it onto the image plane of $(I_i, D_i)$. For-
mally,

$$x' = g_{uv}(s(h(x, D_j(x)), T)).$$

(2)

Here $h$ is the conversion from a depth pixel to a 3D point in
homogenous coordinates:

$$h(u, v, d) = \left(\frac{(u - c_x) \cdot d}{f_x}, \frac{(v - c_y) \cdot d}{f_y}, d, 1\right)^T,$$

(3)

where $f_x$ and $f_y$ are the focal lengths and $(c_x, c_y)$ is the
principal point. $s$ is the following rigid transformation:

$$s(h, T) = Th.$$

(4)

$g$ is the inverse function of $h$, which maps a 3D point to a
depth pixel:

$$g(s_x, s_y, s_z, 1) = \left(\frac{s_x f_x}{s_z} + c_x, \frac{s_y f_y}{s_z} + c_y, s_z\right)^T.$$

(5)
The first two components of \( g \), denoted by \( g_{uv} \), form the corresponding pixel \( x' \) on the image plane of \((I_i, D_i)\).

Similarly, we can define a geometric objective \( E_D \) that compares the depth of pixel \( x \) and \( x' \). We notice that direct comparison between depth values \( D_i(x') \) and \( D_j(x) \) leads to incorrect results since the depth values are measured in different camera spaces. We therefore compare \( D_i(x') \) with the warped depth \( g_d \), which is the third component of \( g \) as defined in Equation 5:

\[
E_D(T) = \sum_x (D_i(x') - g_d(s(h(x, D_j(x)), T)))^2. \tag{6}
\]

It is important that both the photometric objective \( E_I \) and the geometric objective \( E_D \) are defined on the same parameterization domain. In the next section, we show that a change of parameterization domain enables generalization of these objectives to unstructured point clouds.

A joint photometric and geometric objective can be formulated by combining \( E_I \) and \( E_D \):

\[
E(T) = (1 - \sigma)E_I(T) + \sigma E_D(T), \tag{7}
\]

where \( \sigma \in [0, 1] \) is a constant weight that balances the two terms.

4. Colored Point Cloud Registration

In this section we generalize the joint optimization objective (7) to aligning colored point clouds.

4.1. Parameterization

Let \( P \) be a colored point cloud, and let \( C(p) \) be a discrete function that retrieves the intensity of each point \( p \). In order to use color in optimization, we need to generalize \( C(p) \) to a continuous function so that we can compute its gradient.

Conceptually, we introduce a virtual orthogonal camera for each point \( p \in P \). It is configured to observe \( p \) along the normal \( n_p \). The image plane of this virtual camera is the tangent plane at \( p \). It parameterizes a virtual image that can be represented as a continuous color function \( C_p(u) \), where \( u \) is a vector emanating from \( p \) along the tangent plane: \( u \cdot n_p = 0 \). The function \( C_p(u) \) can be approximated by its first-order approximation:

\[
C_p(u) \approx C(p) + d_p \top u, \tag{8}
\]

where \( d_p \) is the gradient of \( C_p(u) \). The gradient is estimated by applying least-squares fitting to \( \{C(p')|p' \in N_p\} \), where \( N_p \) is the local neighborhood of \( p \).

Specifically, let \( f(s) \) be the function that projects a 3D point \( s \) to the tangent plane of \( p \):

\[
f(s) = s - n_p(s - p) \top n_p. \tag{9}
\]

The least-squares fitting objective for computing \( d_p \) is

\[
L(d_p) = \sum_{p' \in N_p} (C_p(f(p') - p) - C(p'))^2
\approx \sum_{p' \in N_p} (C(p) + d_p \top (f(p') - p) - C(p'))^2, \tag{10}
\]

with the additional constraint \( d_p \top n_p = 0 \). This is a linear least-squares problem and can be solved efficiently during preprocessing.

Similarly, we can assume that the virtual camera has a depth channel and define a continuous depth function \( G_p(u) \). Since its gradient at the origin is 0, the first-order approximation of \( G_p(u) \) is a constant function:

\[
G_p(u) \approx (o_p - p) \top n_p, \tag{11}
\]

where \( o_p \) is the origin of the virtual camera.

4.2. Objective

Let \( P \) and \( Q \) be two colored point clouds and let \( T^0 \) be the coarse initial alignment. Our goal is to find the optimal transformation \( T \) that aligns \( Q \) to \( P \).

We formulate a joint optimization objective

\[
E(T) = (1 - \sigma)E_C(T) + \sigma E_G(T), \tag{12}
\]

where \( E_C \) and \( E_G \) are the photometric and geometric terms, respectively. \( \sigma \in [0, 1] \) is a weight that balances the two terms.

The term \( E_C \) is defined by generalizing the photometric term \( E_I \) in Equation 1. The first change we make is to define residuals based on a correspondence set \( K = \{(p, q)\} \) instead of the pixel set \( \{x\} \). Here \( K \) is created following the ICP algorithm: in each optimization iteration, \( K \) is recomputed as the set of correspondence pairs between \( P \) and \( T^kQ \) that are within distance \( \varepsilon \), where \( T^k \) is the current transformation.

To use the virtual camera introduced in Section 4.1, \( q \) is projected to a point \( q' \) on the tangent plane of \( p \):

\[
q' = f(s(q, T)), \tag{13}
\]

where \( s \) is the rigid transformation in Equation 4 and \( f \) is the projection function in Equation 9. Using the local color function \( C_p \) in (8) and the projected point \( q' \) in (13), \( E_C \) is defined as

\[
E_C(T) = \sum_{(p, q) \in K} (C_p(q') - C(q))^2. \tag{14}
\]

Similarly, we generalize the geometric term \( E_D \) in Equation 6 to \( E_G \):

\[
E_G(T) = \sum_{(p, q) \in K} (G_p(q') - (o_p - s(q, T)) \top n_p)^2. \tag{15}
\]
Substituting \( G_p(q') \) using (11), variable \( \mathbf{d}_p \) is eliminated:

\[
E_G(T) = \sum_{(p,q) \in \mathcal{K}} \left( (s(q, T) - \mathbf{p})^\top \mathbf{n}_p \right)^2.
\] (16)

This function is equivalent to the point-to-plane objective in the ICP algorithm [3, 31]. When only the geometric term is used \((\sigma = 1)\), our algorithm reduces to point-to-plane ICP.

Putting everything together, the joint optimization objective (12) can be written as

\[
E(T) = (1 - \sigma) \sum_{(p,q) \in \mathcal{K}} \left( r_C^{(p,q)}(T) \right)^2 + \sigma \sum_{(p,q) \in \mathcal{K}} \left( r_G^{(p,q)}(T) \right)^2,
\] (17)

where \( r_C^{(p,q)} \) and \( r_G^{(p,q)} \) are the photometric and geometric residuals, respectively:

\[
r_C^{(p,q)}(T) = C_p(f(s(q, T))) - C(q),
\] (18)

\[
r_G^{(p,q)}(T) = (s(q, T) - \mathbf{p})^\top \mathbf{n}_p.
\] (19)

### 4.3. Optimization

We minimize the nonlinear least-squares objective \( E(T) \) using the Gauss-Newton method. In each iteration, we linearize \( T \) locally as a 6-vector \( \xi = (\alpha, \beta, \gamma, a, b, c) \), which collates a rotational component \( \omega \) and a translation \( \mathbf{t} \). \( T \) is approximated by a linear function of \( \xi \):

\[
T \approx \begin{pmatrix} 1 & -\gamma & \beta & a \\ \gamma & 1 & -\alpha & b \\ -\beta & \alpha & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix} T^k,
\] (20)

where \( T^k \) is the transformation estimated in the last iteration. Following the Gauss-Newton method, we compute \( \xi \) by solving the linear system

\[
J_r^\top J_r \xi = -J_r^\top r,
\] (21)

where \( r \) is the residual vector and \( J_r \) is its Jacobian, both evaluated at \( T^k \):

\[
r = \left[ \sqrt{1 - \sigma r_C}; \sqrt{\sigma} r_G \right],
\] (22)

\[
r_C = \left[ r_C^{(p,q)}(T)|_{T=T^k} \right]_{(p,q)},
\] (23)

\[
r_G = \left[ r_G^{(p,q)}(T)|_{T=T^k} \right]_{(p,q)},
\] (24)

\[
J_r = \left[ \sqrt{1 - \sigma} J_{r_C}; \sqrt{\sigma} J_{r_G} \right],
\] (25)

\[
J_{r_C} = \left[ \nabla r_C^{(p,q)}(T)|_{T=T^k} \right]_{(p,q)},
\] (26)

\[
J_{r_G} = \left[ \nabla r_G^{(p,q)}(T)|_{T=T^k} \right]_{(p,q)}. \] (27)

To evaluate the partial derivatives in Equations 26 and 27, we use (18) and (19) and apply the chain rule:

\[
\nabla r_C^{(p,q)}(T) = \frac{\partial}{\partial \xi}(C_p \circ f \circ s) = \nabla C_p(f(s)) \mathbf{J}_f(s) \mathbf{J}_s(\xi),
\] (28)

\[
\nabla r_G^{(p,q)}(T) = \mathbf{n}_p^\top \mathbf{J}_s(\xi),
\] (29)

where \( \nabla C_p = \mathbf{d}_p \) is the precomputed gradient for each point \( p \in \mathbf{P} \), \( \mathbf{J}_f(s) \) is the Jacobian of \( f \) derived from (9), and \( \mathbf{J}_s \) is the Jacobian of \( s \) with respect to \( \xi \), derived from (4) and (20).

In each iteration, we evaluate the residual \( r \) and the Jacobian \( J_r \) at \( T^k \), solve the linear system in (21), update \( T \) by applying the incremental transformation \( \xi \) to \( T^k \) using (20), and map the transformation into \( SE(3) \). In the next iteration, we reparameterize \( T \) around \( T^{k+1} \) and repeat.

#### 4.4. Coarse-to-fine processing

Objective (12) is non-convex and the optimization can get trapped in local minima. To alleviate this problem, we use a coarse-to-fine scheme. We build a point cloud pyramid by downsampling the input point cloud using a voxel grid with increasing voxel size. The downsampling algorithm approximates the points in each voxel with their centroid. Therefore, in terms of the optimization objective, a residual at a coarser level is the combination of several residuals at a finer level. The objective function at a coarser level is smoother and can guide the Gauss-Newton method to deeper minima. The optimization is performed at each level of the pyramid, from coarse to fine. The result of a coarse level initializes the optimization at the next level.

Algorithm 1 summarizes the presented algorithm for colored point cloud registration.

### 5. Scene Reconstruction

We have presented joint photometric and geometric optimization algorithms for aligning RGB-D images (Section 3) and colored point clouds (Section 4). The benefit of the joint objective is that it locks the alignment both along the tangent plane (via the photometric term) and along the normal direction (via the geometric term). Thus it is more robust and more accurate than using either objective alone. We now demonstrate the utility of these algorithms by using them to increase the accuracy and robustness of a state-of-the-art scene reconstruction system [4].

We build on the publicly available implementation of this system, replacing two key steps using the algorithms presented in this paper. The system takes an RGB-D sequence as input and proceeds through the following steps:

1. Build local geometric surfaces \( \{ P_i \} \) (referred to as fragments) from short subsequences of the input RGB-D sequence;
Algorithm 1 Colored point cloud alignment

**Input:** Colored point cloud \( P \) and \( Q \), initial transformation \( T^0 \)

**Output:** Transformation \( T \) that aligns \( Q \) to \( P \)

1. Build point cloud pyramids \( \{ P^l \} \) and \( \{ Q^l \} \)
2. for \( p \in P^l \) do
3. Precompute \( d_p \) by minimizing (10)
4. This defines function \( C_p \)
5. \( T \leftarrow T^0 \), \( L \leftarrow \max \text{pyramid level} \)
6. for \( l \in \{ L, L = 1, \ldots, 0 \} \) do
7. while not converged do
8. \( r \leftarrow 0, J_r \leftarrow 0 \)
9. Compute the correspondence set \( K \)
10. for \( (p, q) \in K \) do
11. Compute \( r^{(p,q)}, r^{(q,p)} \) at \( T \) (Eq. 18,19)
12. Compute \( \nabla r^{(p,q)}, \nabla r^{(q,p)} \) at \( T \) (Eq. 29,30)
13. Update \( r \) and \( J_r \) accordingly
14. Solve linear system 21 to get \( \xi \)
15. Update \( T \) using Equation 20, then map to \( SE(3) \)
16. Validate if \( T \) aligns \( Q \) to \( P \)

2. Perform global registration and detect matchable fragment pairs \( \{(P_i, P_j)\} \) by applying robust graph optimization over global registration results;
3. Tightly align matchable fragment pairs \( \{(P_i, P_j)\} \) and build correspondence sets \( \{K_{i,j}\} \) between matchable fragments;
4. Optimize the fragment poses \( \{T_i\} \) and a camera calibration function \( C(\cdot) \) by minimizing an objective defined over the correspondences \( \{K_{i,j}\} \) [45];
5. Integrate RGB-D images to generate a mesh model for the scene.

We use the algorithms presented in Sections 3 and 4 to replace Steps 1 and 3.

**Better fragment construction.** We create a fragment from every \( k = 100 \) RGB-D images. Within each subsequence, we test every pair of RGB-D images to see if they can be aligned. The initial alignment is estimated by building correspondences between ORB features in the color images [30], pruning with the 5-point RANSAC algorithm [36], and computing a transformation that aligns the corresponding depth pixels [9]. We then optimize objective (7) to obtain a tight alignment. The algorithm is detailed in the supplement. The alignment results are treated as edges in a pose graph. Robust pose graph optimization is performed to estimate the camera pose of each RGB-D image [4]. With a truncated signed distance volume and a color volume, RGB-D images are integrated into fragments in the form of colored point clouds [6,29,40]. This replaces Step 1.

**Better fragment alignment.** For tight alignment of fragment pairs, we use the colored point cloud alignment algorithm developed in Section 4. This provides more accurate fragment alignment. In particular, the new algorithm is considerably more robust to slippage along flat surfaces, as shown in Figure 1. This replaces Step 3.

6. Dataset

To our knowledge, no publicly available RGB-D dataset provides dense ground-truth surface geometry across large-scale real-world scenes. To complement existing datasets, we have created ground-truth models of five complete indoor environments using a high-end laser scanner, and captured RGB-D video sequences of these scenes. This data enables quantitative evaluation of real-world scene reconstruction and will be made publicly available.

We scanned five scenes: Apartment, Bedroom, Boardroom, Lobby, and Loft. The size of each scene ranges from 21 to 86 square meters. Ground-truth data was collected using a FARO Focus 3D X330 HDR scanner. The scanner has an operating range of 0.6m to 330m. At a distance of 10 meters, its ranging accuracy is 0.1 millimeters. Each scene was scanned from multiple locations. The scans were merged using dedicated software provided by the manufacturer, which is used for range scan alignment in industrial applications.

In each scene, we captured a continuous RGB-D video sequence using an Asus Xtion Live camera. The lengths of the sequences range from 11 to 18 minutes. Each sequence thoroughly covers the respective scene. The RGB-D sequences can be used as input to scene reconstruction systems. The ground-truth models can be used to evaluate the accuracy of the results. The dataset is summarized in Table 1. The ground-truth models are visualized in the supplement.
Figure 2. Evaluation of point cloud alignment on the TUM RGB-D dataset [37]. The presented algorithm is compared to prior algorithms that use color (top) and to algorithms that do not (bottom). The algorithms are initialized with transformations that are perturbed away from the true pose in the rotational component (left) and the translational component (right). The plot shows the median RMSE at convergence (bold curve) and the 40%-60% range of RMSE across trials (shaded region). Lower is better. Our algorithm outperforms all prior methods.

<table>
<thead>
<tr>
<th>Name</th>
<th>Size (m$^2$)</th>
<th># of laser pts.</th>
<th># of RGBD frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apartment</td>
<td>69.17</td>
<td>18.7M</td>
<td>31.9K</td>
</tr>
<tr>
<td>Bedroom</td>
<td>21.01</td>
<td>10.9M</td>
<td>21.9K</td>
</tr>
<tr>
<td>Boardroom</td>
<td>60.90</td>
<td>17.4M</td>
<td>24.3K</td>
</tr>
<tr>
<td>Lobby</td>
<td>86.46</td>
<td>14.5M</td>
<td>20.0K</td>
</tr>
<tr>
<td>Loft</td>
<td>34.74</td>
<td>14.5M</td>
<td>25.3K</td>
</tr>
</tbody>
</table>

Table 1. Dataset statistics. Dense ground-truth surface models were acquired for five real-world scenes using an industrial laser scanner. The scenes were then scanned with an RGB-D camera.

7. Results

7.1. Colored point cloud registration

We begin by evaluating the colored point cloud registration algorithm presented in Section 4. We compare the algorithm to three alternative algorithms for colored point cloud alignment. The first two are ICP variants that embed the point clouds in a higher-dimensional space: the algorithm of Men et al. [28] (referred to as Color 4D ICP) and the algorithm of Johnson et al. [21] (referred to as Color 6D ICP). The third is the algorithm of Korn et al. [27], as implemented in the Point Cloud Library [32] (referred to as Generalized 6D ICP).

The first evaluation was performed on four sequences from the TUM RGB-D dataset: fr1/desk, fr1/desk2, fr1/room, and fr3/office [37]. We split the RGB-D sequences into segments and construct colored fragments using volumetric integration [6] with the ground truth camera poses provided in the dataset. This gives us colored point clouds with known relative poses. We tested registration algorithms on pairs of point clouds that overlap by at least 30%. To evaluate the accuracy of the different algorithms as a function of the initial pose, we initialized them in two regimes. In the first, the rotational component of the initial transformation was perturbed away from the true pose. In the second, the translational component was perturbed. The results are shown in Figure 2 (top). Our algorithm is more accurate when the initialization is near the true pose, and is much more robust to poor initialization.

For completeness, we also evaluate against registration algorithms that do not use color. The results are shown in Figure 2 (bottom). PCL ICP is the Point Cloud Library implementation of the ICP algorithm [32]. Sparse ICP is the algorithm of Bouaziz et al. [2]. We tested these algorithms with both point-to-point and point-to-plane distance measures [31]. Generalized ICP is a Point Cloud Library implementation of the algorithm of Segal et al. [34]. FGR is the state-of-the-art global registration algorithm of Zhou et al. [47]. Our geometric term refers to our results using only the geometric term ($\sigma = 1$). Ours refers to our results using the full optimization objective. The difference between Ours and Our geometric term shows the benefit of using color information. The optimal value of $\sigma$ is found by grid search, as detailed in the supplement.

Our second evaluation was conducted on the Cathedral scene from the multimodal IMPART dataset [26]. The dataset provides seven colored LiDAR scans of a large out-
door scene, captured by a FARO laser scanner. The density of the points is not uniform. The results on this dataset are analogous to Figure 2 and are provided in the supplement. If the initial perturbation is more than 35 degrees in rotation or 4 meters in translation, the other methods begin to fail (RMSE > 0.25m). In contrast, our algorithm aligns the scans tightly even when the initial perturbation is 40 degrees in rotation and 6 meters in translation.

The running time of the different algorithms is reported in Table 2. Runtime was measured on an Intel Core i7-5960X CPU with 8 parallelized threads. Our algorithm is faster than all other local registration algorithms. We hypothesize that the optimization converges faster due to the coarse-to-fine scheme and the photometric term.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color 4D ICP [28]</td>
<td>3.64</td>
</tr>
<tr>
<td>Color 6D ICP [21]</td>
<td>3.66</td>
</tr>
<tr>
<td>Generalized 6D ICP [27]</td>
<td>16.11</td>
</tr>
<tr>
<td>Generalized ICP [34]</td>
<td>3.54</td>
</tr>
<tr>
<td>PCL ICP (point to point) [32]</td>
<td>2.43</td>
</tr>
<tr>
<td>PCL ICP (point to plane) [32]</td>
<td>1.77</td>
</tr>
<tr>
<td>Sparse ICP (point to point) [2]</td>
<td>8.96</td>
</tr>
<tr>
<td>Sparse ICP (point to plane) [2]</td>
<td>9.41</td>
</tr>
<tr>
<td>FGR * [47]</td>
<td>0.37</td>
</tr>
<tr>
<td>Ours</td>
<td>0.70</td>
</tr>
</tbody>
</table>

Table 2. Average running time (seconds). *FGR is a global registration algorithm that operates on fixed correspondences.

7.2. Scene reconstruction

We now evaluate the enhanced scene reconstruction system described in Section 5. Our first baseline is the system of Choi et al. [4] without our enhancements (referred to as CZK). Our second baseline is the ElasticFusion system of Whelan et al. [41], a state-of-the-art real-time pipeline. Note that neither our system nor CZK operate in real time, so ElasticFusion is at a disadvantage.

We begin with an evaluation on the existing SceneNN dataset [19]. This dataset does not provide ground-truth models, so our evaluation here is qualitative. We randomly sample two sequences from the dataset and reconstruct them with the three pipelines. The results are shown in Figure 4. For the purpose of visualization, Poisson surface reconstruction is applied to the output of ElasticFusion to create a mesh [23]. Our system produces the best qualitative results on both randomly sampled scenes.

We now perform a quantitative evaluation on the dataset presented in Section 6. Let the precision of a reconstructed model be the percentage of reconstructed points that have a ground-truth point within distance $\tau$. Let the recall of a reconstructed model be the percentage of ground-truth points that have a reconstructed point within distance $\tau$. We use $\tau = 20$ millimeters. Our primary measure is the F-score, the harmonic mean of precision and recall:

$$F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$  

The F-score achieved by each system on each of the five scenes is reported in Table 3. Our system achieves an average F-score of 59.69%, versus 46.49% achieved by the CZK baseline. The reconstructions produced by our system are visualized in Figure 5.

8. Conclusion

We revisited the problem of colored point cloud registration and presented an algorithm that optimizes a joint
photometric and geometric objective. Our formulation unifies RGB-D image alignment and colored point cloud registration. Our approach outperforms prior registration algorithms. As an application, we used the presented approach to significantly improve the accuracy of a state-of-the-art scene reconstruction system. To quantitatively validate the results on real-world data, we created a dataset of five indoor scenes with accurate ground-truth models. Our dataset and reference implementations will be made publicly available.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Apartment</td>
<td>7.36</td>
<td>55.63</td>
<td><strong>61.68</strong></td>
</tr>
<tr>
<td>Bedroom</td>
<td>13.21</td>
<td>46.17</td>
<td><strong>75.25</strong></td>
</tr>
<tr>
<td>Boardroom</td>
<td>16.41</td>
<td>49.41</td>
<td><strong>50.43</strong></td>
</tr>
<tr>
<td>Lobby</td>
<td>7.35</td>
<td>35.37</td>
<td><strong>48.02</strong></td>
</tr>
<tr>
<td>Loft</td>
<td>30.60</td>
<td>45.88</td>
<td><strong>63.05</strong></td>
</tr>
<tr>
<td>Mean</td>
<td>14.99</td>
<td>46.49</td>
<td><strong>59.69</strong></td>
</tr>
</tbody>
</table>

Table 3. Results on the presented dataset. F-score in percentage points.
References

[27] M. Korn, M. Holzkothen, and J. Pauli. Color supported Generalized-ICP. In *VISAPP*, 2014. 1, 2, 6, 7


