Supplementary material

This is a supplementary material for the paper, *Facial Depth and Normal Estimation using Single Dual-Pixel Camera*. We will further describe details: ground truth acquisition process of our facial DP dataset (Sec. A), concrete description of evaluation metrics (Sec. B), precise algorithm of our Adaptive Normal Module (Sec. C), and generalization experiments using our network and dataset (Sec. D).

A Ground-Truth Acquisition

In this section, we describe the precise ways of ground truth depth acquisition processes from our hardware setup. These processes consist of three sub-processes: (1) depth from structured light (Sec. A.1) (2) multi-view depth refinement (Sec. A.2) (3) normal-assisted depth refinement (Sec. A.4). To this end, we explain detail of our process to acquire parameters of Eq. 1 in manuscript.

A.1 Depth from Structured Light

Under the carefully designed hardware setup, we start to acquire initial ground truth depth maps from structured light. We would like to briefly explain detail processes to acquire unwrapped phase images from coded patterns. As in [24,23], we use 12 horizontal patterns and 10 vertical patterns consisting of 6-bit inverse gray-code (Fig. 13(a),(b)) to get unwrapped phase images, and 8 phase-shifting patterns [19,55] (Fig. 13(c)) for phase correction [30]. These pattern images are used to acquire two (horizontal/vertical) unwrapped phase images per camera view. Then, we estimate dense correspondences based on a standard phase unwrapping method [55]. After that, consistency between multi-view unwrapped phase images’ intensity is used to get high quality of facial depth following [23].

A.2 Multi-View Depth Refinement

Structured light can give us high-quality facial geometry that can be regarded as the ground-truth depth maps. There still exists outliers that mainly comes from small movement of face while capturing different coded patterns. To resolve this problem, we check the visibility of acquired points in perspective view of each camera and remove outliers of gathered point clouds with considering neighbor points. Finally, we check each points’ multi-view photometric/depth consistency (each point should be visible from more than two sampled views) to determine whether the point is inlier or not. Results in Fig. 14 shows that this refinement process can reduce outliers effectively without losing inliers.

A.3 Surface Normal from Photometric Stereo

In this section, we will explain our progress to get good quality of normal map to be used for supervised signal in training step. We use a photometric stereo
as reported in [49] to estimate surface normal of the subjects’ face. First, we capture multiple images with light from varying directions. We calibrate the lighting directions using a chrome ball and estimate them as follows:

\[
L = 2(n \cdot R)N - R, \text{ where } R = (0, 0, 1)^T, n = \frac{1}{r} (n_x, n_y, n_z),
\]

\[
n_x = h_x - c_x, n_y = h_y - c_y, n_z = \sqrt{r^2 - n_x^2 - n_y^2},
\]

where \(c\) and \(h\) are the center of a chrome ball, and the center of specular reflection, respectively. \(r\) is the norm of the normal vector \((n_x, n_y, n_z)\). By finding the distance between the centers \(c\) and \(h\), the lighting direction \(L = (L_x, L_y, L_z)\) can be calculated using Eq. 6.
Facial Depth and Normal Estimation using Single Dual-Pixel Camera

Then, we compute the surface normal of the subject by solving a large over-constrained linear system as below:

\[
\begin{bmatrix}
N_1 \\
\vdots \\
N_p
\end{bmatrix}
= 
\begin{bmatrix}
I_1 \\
\vdots \\
I_p
\end{bmatrix}
\begin{bmatrix}
L_x, L_y, L_z
\end{bmatrix}^\top, \quad p \in P,
\] (7)

Finally, we follow outlier-robust scheme \cite{48,27} to reject non-Lambertian observations by regarding them as outliers.

A.4 Normal-guided Depth Refinement

We further continue to remove outliers and improve the quality of depth by constraining the surface gradient with given normal map from Sec. A.3, which is orthogonal to the photometric normal while keeping its original position as possible. For this, we formulate an energy function consisting of two error terms: the position error \(E_d\) and the normal error \(E_n\):

\[
E = \lambda E_d + (1 - \lambda) E_n,
\]

\[
E_d = \| X_p - X_p^m \|,
\]

\[
E_n = \sum_p \left( [T_x (X_p) \cdot N_p]^2 + [T_y (X_p) \cdot N_p]^2 \right)
\] (8)

where \(X_p\), \(T_{x,y}(X_p)\) are a 3D point and its surface gradient in the \(x, y\)-direction, respectively. \(\lambda\) is the balancing term between the position and the normal errors. The entire minimization is also formulated as an over-constrained linear system to be solved by least squares as described in \cite{49}.

As shown in the Fig. 15, the refinement process effectively removes outliers and improves the overall quality of facial geometry by alleviating the line-artifact and noisy 3D points.

A.5 Conversion from Disparity to Metric Depth

Given the estimated defocus-disparity from our proposed network, StereoDP-Net, we provide exact conversion between the disparity and the metric depth.
Fig. 16. Facial reconstruction from given facial DP image. Given DP input of (a), our proposed StereoDPNet estimates disparity map and surface normal in (b). We show the reconstructed point cloud using correct/incorrect conversion of Sec. A.5 in (c), (d) to show the importance of our calibration process.

Fig. 17. Defocus-disparity to metric depth. (a) We first acquire defocus-disparity and depth obtained by a plane homography. (b) Using the acquired depth and disparity, we find parameters of Eq. 1.

In Sec. 4.3 in the manuscript. As shown in Fig. 16 finding the correct relationship is critical to facial 3D reconstruction since the wrong conversion can result twisted shape. Here, we explain our calibration process in details that covers conversion between a defocus-disparity and a metric-scale depth.

In Fig. 17 (a), we compute the corresponding pair of points of left/right DP images. In particular, we adopt the saddle point refinement method [25] that is robust to defocus blur in DP images. By doing so, we obtain plane depth and defocus-disparity at each point.

In Fig. 17 (b), following Eq. 1 we use the linear relation between the obtained inverse depth and defocus-disparity measurements. To find calibration parameters, we first compute the bias $B(L, f, g)$ and the slope $A(L, f, g)$ through least-square optimization. Then, we obtain the focus distance $g = -\frac{A(L, f, g)}{B(L, f, g)}$. Using focal length $f$ and F number that are pre-defined by the lens condition, we calculate the aperture $L$. $\alpha$ is acquired from the slope $A(L, f, g)$. Finally, we get all the calibration parameters of Eq. 1.
B Evaluation Metrics

Depth Metrics. Previous studies \cite{16,51} predict depth with affine ambiguity. Therefore, their methods only provide experimental results with affine invariant metrics. Following \cite{16}, we measure the quality of affine transformed depth as:

- **Affine invariant metrics**
  - $\text{AIWE}(p) : \min_{a,b} \left( \sum_{u=1}^{W} \sum_{v=1}^{H} \frac{|d_{u,v} - (ad_{u,v} + b)|^p}{|H\cdot W|} \right)^{1/p}$
  - $\text{WMAE} = \text{AIWE}(1)$
  - $\text{WRMSE} = \text{AIWE}(2)$

Furthermore, based on our calibration parameters (Sec. A.5), we are able to measure the accuracy of absolute-scale depth as:

- **Absolute metrics**
  - $\text{RMSE} : \frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} \left( Z_{u,v} - \hat{Z}_{u,v} \right)^2$
  - $\text{AbsRel} : \frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} \left( \frac{Z_{u,v} - \hat{Z}_{u,v}}{Z_{u,v}} \right) |Z_{u,v}|$
  - $\text{MAE} : \frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} |Z_{u,v} - \hat{Z}_{u,v}|$
  - $\delta^i : \frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} \left( \max \left( \frac{Z_{u,v} - \hat{Z}_{u,v}}{Z_{u,v}}, \frac{\hat{Z}_{u,v} - Z_{u,v}}{Z_{u,v}} \right) < \tau^i \right)$

where $\hat{Z}$ denotes estimated depth and $Z$ denotes ground-truth depth. Here, we used $\tau$ as 1.01 where $i \in \{1, 2, 3\}$.

Normal Metrics. Following \cite{38}, we use Mean Angular Error (MAE) and Root Mean Square Angular Error (RMSAE) as:

- **Normal metrics**
  - $\text{MAE} : \frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} \arccos (\hat{n}_{u,v} \cdot \hat{n}_{u,v})$
  - $\text{RMSAE} : \sqrt{\frac{1}{|H\cdot W|} \sum_{u=1}^{W} \sum_{v=1}^{H} \arccos (\hat{n}_{u,v} \cdot \hat{n}_{u,v})^2}$

\footnote{http://www.cvlibs.net/datasets/kitti/}
C Details of Adaptive Normal Module

In this paper, we propose two sub-modules for predicting depth and surface normal from a single pair of facial DP images. Here, we describe details of surface sampling layer of our proposed ANM in Alg. 1 and Alg. 2.

**Algorithm 1** Surface Sampling in ANM

**Require:** Aggregated Cost Volume, $C_A \in \mathbb{R}^{C \times M \times H \times W}$
- Inferred disparity, $\hat{d} \in \mathbb{R}^{H \times W}$
- Number of sampled neighbors, $P=4$
- Number of hypothesis planes in $C_A$, $M=8$

1: **procedure** `SurfaceSample($C_A$, $\hat{d}$, $P$, $M$)

2: $\tilde{d}_P \leftarrow $ Convert-Disparity-to-VolumeIndex($\hat{d}$, $M$) \hfill \triangleright \text{Alg. 2}

3: $C_S \leftarrow $ Sample-$P$-Closest-Neighbors($C_A$, $\tilde{d}_P$, $P$, $M$)

4: return Sampled volume $C_S \in \mathbb{R}^{C \times P \times H \times W}$

5: **end procedure**

**Algorithm 2** Ray Sampling in Volume

**Require:** Inferred disparity $\hat{d} \in \mathbb{R}^{H \times W}$
- Number of hypothesis planes in $C_A$, $M=8$
- $d_{\text{min}} = d^0$, $d_{\text{max}} = d^m$ (Eq. 2 of the manuscript)
- VolumeIndex $\tilde{d}_P \in \mathbb{R}^{3 \times H \times W}$

1: **procedure** `Convert-Disparity-to-VolumeIndex($\hat{d}$, $M$)

2: for $(u, v)$ in $\hat{d}$ do

3: $\tilde{d}_P(u, v) \leftarrow (u, v, \frac{\hat{d}(u, v) - d_{\text{min}}}{d_{\text{max}} - d_{\text{min}}} \cdot M)$

4: end for

5: return VolumeIndex $\tilde{d}_P \in \mathbb{R}^{3 \times H \times W}$

6: **end procedure**
D Supplementary Results

D.1 Comparison of DPNet with the Original

Since there is no public code for DPNet [16], we re-implement DPNet [16] by ourselves following their description. To verify our implementation, we train our DPNet and measure the performance on their dataset [16]. Although there is a little performance drop with our implementation compared to the reported performance in the original paper [16], we show that our implemented model has similar performance with the original implementation in Table 5.

<table>
<thead>
<tr>
<th>Method</th>
<th>Affine error metric ↓</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DPNet (reported in [16])</td>
<td>0.0581 0.0735 0.827</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DPNet (reimplemented)</td>
<td>0.073 0.09 0.883</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5. Comparison of DPNet reported in [16] and re-implemented by ours. We measure the performance of our re-implemented DPNet on their dataset [16] and compare with the reported performance in [16]. Note that we don’t use their affine invariant loss to purely verify the performance of the model.

D.2 Application: Face Relighting

Although our main goal is to estimate the depth and normal from DP images, we introduce naive methods for one of the applications, face relighting, that can be the baseline for future works.

**Face Relighting.** Given the reference images and the surface normals from StereoDPNet, we generate relighted images using a Ratio Image-based method [75]. The target spherical harmonic lightings are randomly sampled and the lighting directions of reference images are inferred with SfSNet [58]. The results are shown in Fig. 18.

D.3 Real-World Results

We show additional real-world results with unmet environment to demonstrate our method and dataset’s generality in Fig. 20 and in Fig. 21 similar to Fig. 8 of manuscript. We also note that our method is able to apply with various camera parameters (focus distance from 1.0m to 1.5m and F-number from 2.0 to 7.1). We also demonstrate that our method can deal with various facial expressions in Fig. 19.
Fig. 18. Face relighting from our estimated surface normal. We display the original DP image, estimated normal map from StereoDPNet, and two different relit images with sampled lighting directions.

Fig. 19. Depth and Normal with facial expressions. We demonstrate that our method can cover face with various facial expressions thanks to our carefully designed facial dataset.
Fig. 20. **Real-world results.** More real-world results with captured camera settings similar to Fig. 5.
Fig. 21. **Real-world results.** More real-world results with captured camera settings similar to Fig. 8.
References


