Optical Non-Line-of-Sight Physics-based 3D Human Pose Estimation
Supplementary Materials

Mariko Isogawa, Ye Yuan, Matthew O'Toole, Kris Kitani
Carnegie Mellon University

Contents

1. Overview of the Supplementary Materials

This supplementary document contains additional details on the NLOS imaging problem and the implementation of our NLOS pose estimation framework. Please also refer to the supplementary video (https://youtu.be/4HFulrdmLE8) for additional information and results. We highlight reference numbers associated with the main paper in blue, and those associated with this supplementary document in red.

2. Non-Line-of-Sight (NLOS) Imaging

2.1. The Property of Transient Images: How to Acquire, How It Looks Like, and Why the Task is Difficult

This section aims to give a better intuition for the property of the transient images and their data acquisition processes. A transient image is a 3D measurement volume, containing a scene's spatio-temporal response to laser pulse. Each voxel encodes the number of photons at a specific (2D) point in space and at a specific (1D) point in time (see Fig. 1(b)). In confocal NLOS imaging [5], the transient image captures light travelling between a specific point (x, y) on a wall, and a hidden scene. As shown in Fig. 1(a), a single pulsed laser and a transient sensor record the time light takes to travel from a point on a wall to the person hidden from the sensor's line of sight. That is, the laser light first travels (i) from the pulsed laser to the visible wall, and the visible wall to the hidden person. Then, (ii) reflected laser from the person goes back to the visible wall, and finally acquired by a co-axial transient sensor. This time of flight data acquisition is repeated for a $n \times n$ grid of point on the wall (*e.g.*, 32×32 points) by raster scanning the surface one point at a time (see green line in Fig. 1(a)).

As shown in Fig. 1(b), for any one point (x, y), a transient measurement is a histogram of the travel time of photons. The location of the peak intensity represents the travel time required for most photons to return to point (x, y), and corresponds to the distance that separates the hidden object from point (x, y). The confocal transient image is a collection of such

measurements for all points on the wall, and has the dimension (x,y,t) = (n,n,b), where b is the number of travel time bins in the histogram. The confocal transient images used in this work [4] were sampled at a resolution of 32×32 spatial points and 4096 time bins. Cross-sections of a single transient image at different instances in time are shown in Fig. 1(c). The transient images show the light reflected by a hidden object, and appearing on a visible surface (*i.e.*, wall) as a function of time.

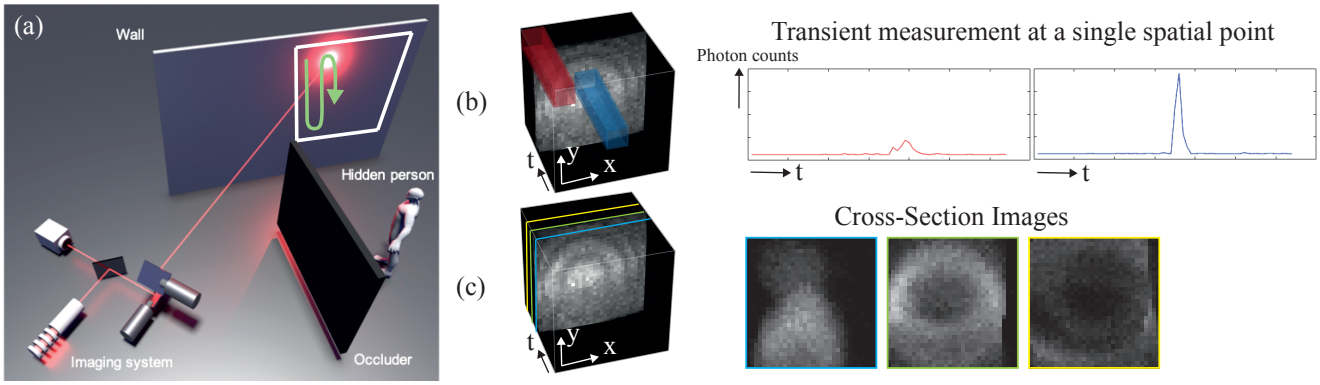

Figure 1. Overview of confocal transient imaging. **(a)** A pulsed laser illuminates the wall at a point and a transient sensor measures the temporal response at the same point. The system then raster scans the wall one point at a time to form a 3D volume of measurements. **(b)** & **(c)** Deciphering meaningful 3D structure from transient images is non-trivial, which makes human pose estimation from transient imaging a difficult problem.

As made evident in Fig. 1 and 2, estimating physically consistent 3D human pose from transient images is challenging, when compared to working with regular RGB and RGBD data. Fig. 2 shows reconstructed human shape with the NLOS imaging algorithm [5] used in our pipeline. The 2D images were generated by taking max intensity of the reconstructed 3D volume given transient images via this method. Even with this state-of-the-art NLOS imaging method, the reconstructed images are noisy, have low spatial resolution, and are recorded at slow frame rates. This makes it difficult to capture small shape details and fast motion, both of which are important factors when estimating a human pose sequence. Furthermore, due to the light lost after multiple scattering events, very few photons reach the sensor and the acquired transient image can therefore be very noisy. All of these characteristics make it very challenging to estimate 3D human pose directly from the transient image. Please note that due to the unique visual properties of transient images, current state-of-the-art human pose estimation methods for RGB image frames (*e.g.*, [1, 2, 8]) cannot be applied directly to these transient images.

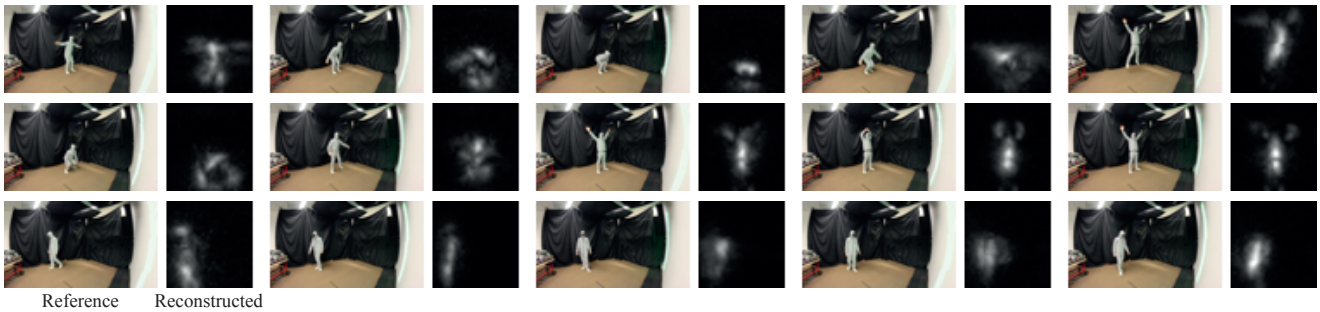

Figure 2. Reconstructed human shape by NLOS imaging [5]. Even with the state-of-the-art NLOS imaging algorithm, the reconstructed results are very blurry, noisy and have both spatially and temporally low resolution, which makes our task (*i.e.*, physics based 3D human pose estimation) quite challenging.

2.2. Pseudo-Transient Images

Sec. 3.2.2 in the main paper introduced our pseudo-transient image synthesis as a training data. We introduced four different types of noises and operations to close the domain gap that exists between pseudo-transient images and real transient images. This section shows visualizations of pseudo-transient images, and also provides additional details on the *temporal resampling* strategy used for data augmentation.

2.2.1 Visualized Results of the Pseudo-Transient Images, and Reconstructed Depth

Fig. 3 highlights both synthesized pseudo-transient images and real transient images of persons sharing a similar pose. To better visualize the volumes, we also show cross-sections (in time) of these transient images. Note that the cross-section containing the highest signal occurs at different times, because this depends on the person's location in the hidden environment. We apply five levels of temporal shift (see the main paper Sec. 3.2.2) to augment the pseudo-transient images, and make our procedure robust to a person's position. Some of the slices also show discontinuities along the horizontal axis; this is an artifact of raster scanning a wall while a person moves within the hidden scene. We reproduce these artifacts for our pseudo-transient images by simulating the same raster scanning procedure; please refer Sec. 2.2.2 for more details. Note that the reference video frames are only for reference, and not used in the pose estimation process. As shown in the figure, our pseudo-transient images closely match the real transient images.

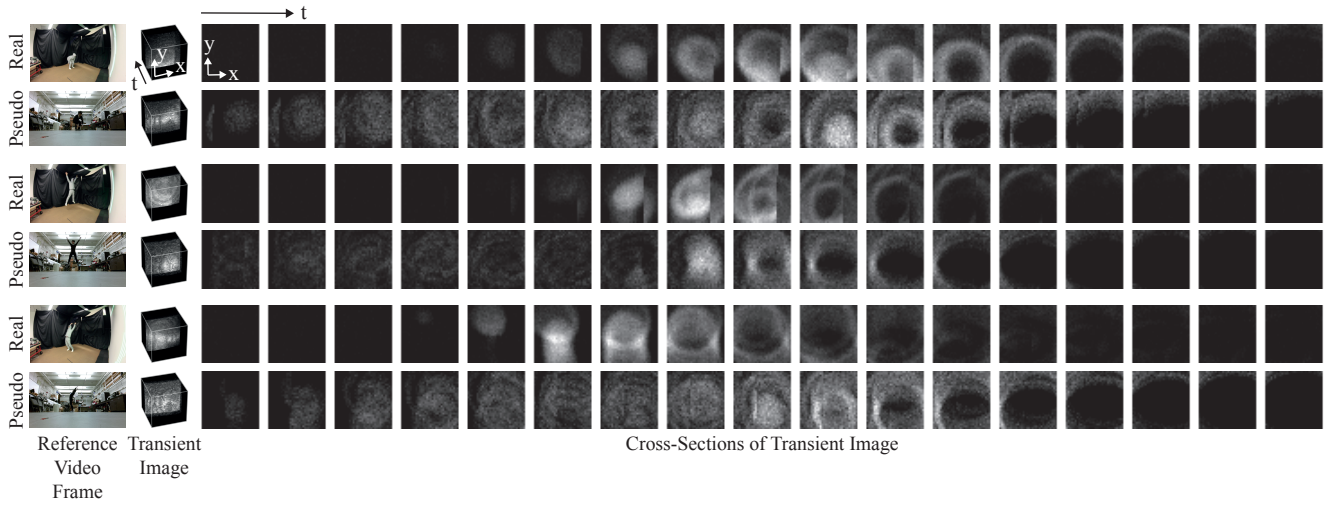

Figure 3. Visualization of the real captured transient images and pseudo-transient images, for persons sharing a similar pose. Note that the reference video frames are only used for comparison purposes; they are not used as input during pose estimation.

2.2.2 Data Augmentation Strategy (Temporal Resampling)

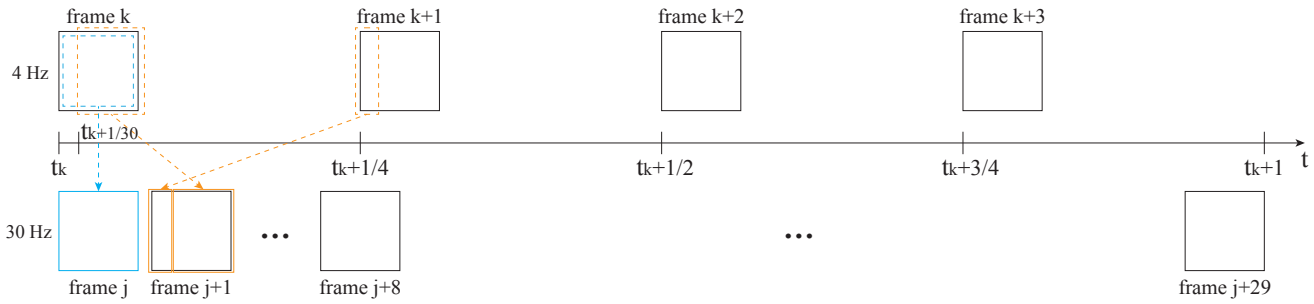

Figure 4. Temporal resampling from 4 Hz to 30 Hz.

Confocal NLOS imaging requires raster scanning a visible surface point by point. For the data used in this work [4], the confocal transient measurements are sampled at 32×32 locations at a frame rate of 4 Hz.

As discussed in Sec. 3.2.2, we introduce a procedure to temporally re-sample transient image sequences. First, to simulate the raster scanning procedure, we capture depth maps at 30 Hz, convert them to pseudo-transient measurements, and simulate raster-scanned measurements by down-sampling the result to 4 Hz. Second, we invert the raster-scanning process, and up-sample the pseudo-transient data from 4 Hz back to 30 Hz.

As shown in the top row of Fig. 4, each transient image is a collection of transients scanned from time t_k to $t_k + \frac{1}{4}$, where $t_k = \frac{k}{4}$ represents the start time of the k^{th} frame (in seconds). To generate a 30 Hz transient sequence (the bottom row), we simply assemble transients captured within the same time range, but set the start time of the k^{th} frame to $t_k = \frac{k}{30}$.

Here, to simplify the explanation, in Fig. 4 we assume j^{th} frame in 30 Hz transient sequence starts at the same start time with k^{th} frame in 4 Hz. Assembling 30 Hz frames is simply combining 4 Hz frames with starting scanning point $t_k + \beta$, where $\beta > 0$ incremented with a constant step $\frac{1}{30}$ sec. For example, since the starting time of k^{th} frame in 4 Hz j^{th} frame in 30 Hz is the same, j^{th} frame in 30 Hz is obtained by just copying k^{th} frame in 4 Hz as the blue lines show. The next $j+1^{\text{th}}$ frame in 30 Hz is assembled with 4 Hz frame scanned from $t_k + \frac{1}{30}$ to $t_k + \frac{1}{4} + \frac{1}{30}$. Thus as orange lines show, copying scanned data in this time range to the same spatial point generates resampled frame $j+1^{\text{th}}$. Converting frames from 30 Hz to 4 Hz is just a inversion of this process.

2.3. Background: Detailed Derivation of the Confocal NLOS Imaging

The confocal NLOS imaging that we use [5] aims to convert 3D transient image $\tau(x', y', t)$ into a discretized reconstruction 3D volume $\rho(x, y, z)$, which represents the reflectivity at every point (x, y, z), $z > 0$ in space, as we show with Eq. 3 in the main paper. Here, this section shows more detailed derivation of the equation.

In the continuous domain, the relationship between a 3D reconstruction volume $\rho(x, y, z)$ and a 3D transient image $\tau(x, y, t)$ are represented as below.

$$\tau(x', y', t) \quad = \quad \iiint_{\Omega} \tfrac{1}{r^4}\, \rho(x, y, z)\, \delta(2\sqrt{(x'-x)^2 + (y'-y)^2 + z^2} - tc)\, dx\, dy\, dz, \tag{1}$$

where c is the speed of light. Eq. 1 shows that a transient measurement sample $\tau(x', y', t)$ captures the photon flux at point (x', y') and time t, relative to an pulse scattered by the same point at time $t = 0$. Ω represents a 3D half-space containing the hidden space on one side of the wall ($z > 0$). The Dirac delta function δ represents a four-dimensional spatio-temporal hypercone surface, represented by $x^2 + y^2 + z^2 - (tc/2)^2$; this hypercone (or light-cone) models light propagation from the visible wall to the object and back to the visible wall. Here, the distance function $r = \sqrt{(x'-x)^2 + (y'-y)^2 + z^2} = tc/2$ in Eq. 1 can be represented with the light arrival time t. Thus, the term $1/r^4$ can be removed from the triple integral. Also, by replacing variables as $z = \sqrt{u}$, $dz/du = 1/(2\sqrt{u})$, and $v = (tc/2)^2$, the Equation 1 can be re-written as

$$\underbrace{v^{3/2}\tau(x', y', 2\sqrt{v}/c)}_{R_t\{\tau\}(x', y', v)} \quad = \quad \iiint_{\Omega} \underbrace{\tfrac{1}{2\sqrt{u}}\, \rho(x, y, \sqrt{u})}_{R_z\{\rho\}(x,y,u)}\, \underbrace{\delta((x'-x)^2 + (y'-y)^2 + z^2 + u - v)}_{h(x'-x, y'-y, v-u)}\, dx\, dy\, du\,, \tag{2}$$

which can be expressed as a 3D convolution $R_t\{\tau\} = h * R_z\{\rho\}$, where $*$ is the 3D convolution operator. h is a known 3D point spread function (PSF), and describes the transient response of a single scatterer. $R_z\{\cdot\}$ resamples ρ along the z-axis and attenuates the result by $1/2\sqrt{u}$, and $R_t\{\cdot\}$ resamples τ along the time axis and scales the result by $v^{\frac{3}{2}}$. With resampled transient image and reconstruction volume as $\tilde{\tau} = R_t\{\tau\}$ and $\tilde{\rho} = R_z\{\rho\}$ respectively, the forward image formation model for confocal NLOS images becomes a simple 3D convolution operation:

$$\tilde{\tau} = h * \tilde{\rho}\,, \tag{3}$$

Also, this convolution can be rewritten with the matrix-form as follows:

$$\tau = R_t^{-1} F^{-1} \hat{H} F R_z \rho\,, \tag{4}$$

where we vectorize volumes τ and ρ. The matrix F represents a 3D discrete Fourier transform and $\hat{H}$ is a diagonal matrix representing the Fourier transform of the PSF h. The NLOS imaging process reconstructs a 3D volume ρ_* from a transient image τ by inverting Eq. 4 and solving a 3D deconvolution procedure (*e.g.*, using the Wiener filter):

$$\rho_* = R_z^{-1} F^{-1} \underbrace{\left[\frac{\hat{H}^*}{|\hat{H}|^2 + \frac{1}{\alpha}}\right]}_{\textbf{the inverse PSF}} F R_t \tau\,, \tag{5}$$

where a user-defined parameter α controls how sensitive the inverse PSF is to noise.

3. DeepRL Based Pose Estimation Pipeline

This section explains implementation details about our DeepRL based 3D human pose estimation pipeline (Sec. 3.1 in the main paper). We use a humanoid model and a physics simulator, and formalize our task of estimating a pose sequence $p_{1:T}$ from a transient image sequence $\tau_{1:T}$ with a Markov Decision process (MDP). The MDP is defined by a tuple $\mathcal{M} = (S, A, P, R, \gamma)$ of states, actions, transition dynamics, a reward function, and a discount factor. At each time step, the humanoid agent samples an action a_t from a policy $\pi(a_t|s_t)$ whose input state s_t contains both the visual context ϕ_t (computed from the transient images) and the humanoid state z_t (*i.e.*, joint angles and velocities). Next, the environment generates the next state s_{t+1} through physics simulation and gives the agent a reward r_t based on how well the humanoid's 3D pose aligns with the ground-truth. Detailed definitions of the state s_t, action a_t, and policy π_θ are given in the main paper.

To solve this MDP, inspired by previous works [6, 10], we apply the Proximal Policy Optimization (PPO) [7] algorithm to obtain the optimal policy π^* that maximizes the expected discounted return $\mathbb{E}[\sum_{t=1}^{T} \gamma^{t-1} r_t]$. In the following, we first describe the design of the reward function r_t in Sec. 3.1, and then explain the policy learning algorithm in Sec. 3.2. Additionally, we briefly describe in Sec. 3.3 the fail-safe mechanism we use to help the humanoid recover from unstable states.

3.1. Reward Function

This section describes the specific reward function we use to train the humanoid policy. Following [10], to encourage the policy to output a pose sequence $p_{1:T}$ that matches the ground-truth $\hat{p}_{1:T}$, we define the reward function as

$$r_t = w_q r_q + w_e r_e + w_p r_p + w_v r_v, \tag{6}$$

where w_p, w_v, w_q, w_e are weighting factors.

The pose reward r_q measures the difference between pose p_t and the ground-truth $\hat{p}_t$ for non-root joints. Let q_t^j and $\hat{q}_t^j$ denote the j-th joint's orientation quaternion of the estimated pose and the ground-truth pose respectively. The pose reward r_q is computed as

$$r_q = \exp\left[-2 \sum_j \left\| q_t^j \ominus \hat{q}_t^j \right\|^2\right]. \tag{7}$$

The end-effector reward r_e evaluates the difference between the local vector of end effector e_t and the ground-truth $\hat{e}_t$. We use head, hands, and feet as end-effectors. The end-effector reward r_e is defined as

$$r_e = \exp\left[-20 \sum_e \left\| e_t - \hat{e}_t \right\|^2\right]. \tag{8}$$

The root pose reward r_p encourages the humanoids root joint to have the same height h_t and orientation quaternion q_t^r as the ground-truth $\hat{h}_t$ and $\hat{q}_t^r$:

$$r_p = \exp\left[-300 \left(\left(h_t - \hat{h}_t \right)^2 + \left\| q_t^r \ominus \hat{q}_t^r \right\|^2 \right)\right]. \tag{9}$$

The root velocity reward r_v penalizes the deviation of the root's linear velocity l_t and angular velocity ω_t^r from the ground-truth $\hat{l}_t$ and $\hat{w}_t$:

$$r_v = \exp\left[-\left\| l_t - \hat{l}_t \right\|^2 - 0.1 \left\| \omega_t^r - \hat{\omega}_t^r \right\|^2\right]. \tag{10}$$

3.2. Policy Learning

To compute the optimal humanoid policy π^* that maximizes the expected discounted return, we use the Proximal Policy Optimization (PPO) [7] algorithm to compute the gradients of the policy π_θ. Traditional policy gradient methods often suffer from catastrophic failure (*i.e.*, the policy π_θ becomes irrecoverably bad) due to noisy policy gradients caused by high variance of the data collected by the policy. To address this problem, PPO uses a mechanism that prevents noisy gradients from changing the policy too much. Specifically, PPO utilizes a clipping function $clip\left(w_t(\theta), 1-\epsilon, 1+\epsilon\right)$ that clips a gradient by setting it to zero whenever the ratio of current/old policies $w_t(\theta)$ is more than ϵ away from 1. Then, PPO minimizes the following loss function:

$$L(\theta) = \mathrm{E}_{s_t, a_t}\left[\min\left(w_t(\theta)\mathcal{A}_t, \mathrm{clip}\left(w_t(\theta), 1-\epsilon, 1+\epsilon\right)\mathcal{A}_t\right)\right], \tag{11}$$

$$w_t(\theta) = \frac{\pi_\theta\left(a_t|s_t\right)}{\pi_{\theta_{old}}\left(a_t|s_t\right)}, \tag{12}$$

where $\mathcal{A}_t$ is an advantage estimate that measures the goodness of taking action a_t in state s_t and $\pi_{\theta_{old}}$ is the old policy before the gradient update with fixed parameters θ_{old}. Our PPO-based policy learning procedure is outlined in Algorithm 1.

Algorithm 1 Policy Learning

Output: parameters θ of the optimal policy π^*
Initialize θ randomly
while not converged **do**
  $s_0 \leftarrow$ sample initial state from the ground truth motion
  for each simulation step t **do**
    $s_t \leftarrow (\phi_t, z_t)$
    $a_t \sim \pi_\theta(a_t|s_t)$
    Apply a and simulate forward one step
    $s_{t+1} \leftarrow$ new state from simulation
    $r_t \leftarrow$ pose matching reward ▷ Reward function: Sec. 3.1
    Record (s_t, a_t, r_t, s_{t+1}) into a memory
  end for
  $\theta_{old} \leftarrow \theta$
  for each update step **do**
    Sample mini-batch of n samples (s_i, a_i, r_i, s_{i+1}) from a memory
    for each (s_i, a_i, r_i, s_{i+1}) **do**
      $\mathcal{A}_i \leftarrow$ compute advantage [7]
      $w_i(\theta) \leftarrow \frac{\pi_\theta(a_i|s_i)}{\pi_{\theta_{old}}(a_i|s_i)}$ ▷ Eq. 12
    end for
    $\theta \leftarrow \theta + \frac{1}{n}\Sigma_i \nabla_\theta \min(w_i(\theta)\mathcal{A}_i, \text{clip}\left(w_i(\theta), 1-\epsilon, 1+\epsilon\right)\mathcal{A}_i)$ ▷ Eq. 11
  end for
end while

3.3. Fail-safe Mechanism and the state regressor $\mathcal{F}$

Sometimes the extreme noise in the transient images can produce irregular control which makes the humanoid fall down to the ground. To prevent this problem, we use the same value function-based fail-safe mechanism as described in [9] to detect unstable humanoid states and reset the humanoid state to the output of the state regressor $\mathcal{F}$. Without using any physics simulation, the state regressor $\mathcal{F}$ directly maps the visual context ϕ_t to the corresponding state z_t with an MLP of two hidden layers (300, 200) and ReLU activations. Using supervised learning, we train $\mathcal{F}$ for 100 iterations with Adam [3] and a learning rate of 1e-4.

4. P2PSF Net Architecture

We show in Fig. 5 a more detailed version of the P2PSF Net (Fig. 3 in the main paper). As discussed in the main paper (Sec. 3.3), P2PSF Net is a volume to volume network introduced as our feature extractor (the network that obtains the transient feature ψ_t from an transient image τ_t). P2PSF Net takes a transient image of resolution $(x, y, t) = 32 \times 32 \times 64$ and outputs a volume $(x, y, d) = 128 \times 64 \times 64$, matching the size of the inverse PSF volume used in confocal NLOS imaging (Eq. 5). The P2PSF Net has nine 3D convolution layers and two residual connections. Please refer to Fig. 5 for the specific size of each layer.

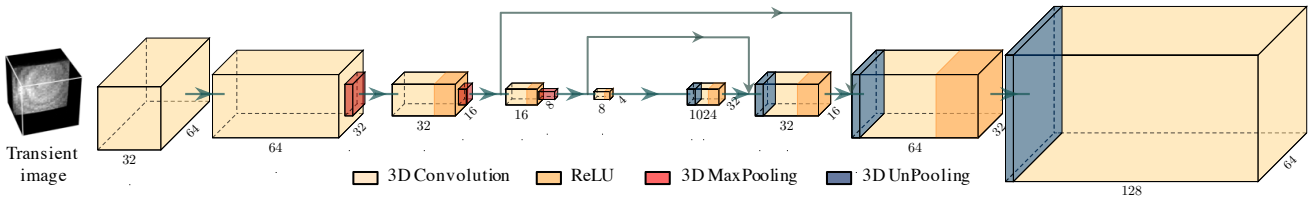

Figure 5. The architecture of P2PSF Net.

References

[1] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint*, 2018. 2

[2] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 2

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 6

[4] David B. Lindell, Gordon Wetzstein, and Matthew O'Toole. Wave-Based Non-Line-of-Sight Imaging using Fast f k Migration. *ACM Transactions on Graphics (TOG)*, 38(4):116, 2019. 2, 3

[5] Matthew O'Toole, David B. Lindell, and Gordon Wetzstein. Confocal non-line-of-sight imaging based on the light-cone transform. *Nature*, 555(7696):338, 2018. 1, 2, 4

[6] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics (TOG)*, 37(4):143:1–143:14, July 2018. 5

[7] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. In *arXiv preprint*, volume abs/1707.06347, 2017. 5, 6

[8] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *British Machine Vision Conference (BMVC)*, 2018. 2

[9] Ye Yuan and Kris Kitani. 3D Ego-Pose Estimation via Imitation Learning. In *European Conference on Computer Vision (ECCV)*, pages 763–778, 2018. 6

[10] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time pd control. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 5