3D RECONSTRUCTION FROM ACCIDENTAL MOTION

- Fisher Yu (Princeton University), David Gallup (Google Inc.)

Team X

Charchit Gupta (2019102034)
Rohith Agaram (2021702026)
Tushar Choudhary (2019111019)

Work Split up

- KLT and experimenting with parameters - Charchit Gupta (2019102034)
- Bundle Adjustment - Tushar Choudhary (2019111019)
- CRF - Rohith Agaram (2021702026)
- Merging KLT + BA + CRF pipelines, debugging and slides preparation - Whole team

The Problem Statement

We tackle the problem of 3D Reconstruction from accidental motion of the photographer. Accidental motion is defined as the inevitable motion that occurs when trying to hold a camera still. Given the initial few frames of a video, we aim to reconstruct a dense depth map of the scene from bundle adjustment which would be given w.r.t. the reference view (frame 1). We further apply a CRF model to regularize the depth to provide a smooth depth map. We demonstrate the results of bundle adjustment for a few scenes.

Steps

The pipeline has the following steps:

Step 1: Extract good features using Shi-Tomasi method.

Step 2: Track the detected features across all the non-reference frames using *Lucas-Kanade*.

Step 3: The tracked features across all the N frames are used to perform the *bundle adjustment* to estimate the 3D structure of the scene.

Step 4: A dense map is reconstructed from the sparse 3D structure using a *CRF model*.

KLT Tracking

We use the KLT tracker to track features across all the N - 1 frames w.r.t the reference frame. This includes two substeps:

(i) Feature detection using Shi-Tomasi: This method uses eigenvalues of the Hessian matrix. The Hessian matrix considers image intensity values in a small square patch in the image, the patch is considered an interest point if min(λ1, λ2) is greater than the threshold value.

(ii) Feature tracking using Lucas-Kanade: This method uses optical flow to estimate the displacement of the points from the non-reference frames (i.e., correspondences) to the reference frame (frame 1).

Structure From Motion

After applying KLT, we have a set of points in each image with known correspondences. To get the depths of these points, we will parameterize the world points in terms of depth, and try to determine these world points this set of image points, which we know is a structure from a motion problem.

Given a set of images depicting a number of 3D points from different viewpoints, the structure from motion can be defined as the problem of simultaneously refining the 3D coordinates of the world points and projection matrix.

Note: In our case, we already knew the camera intrinsics, and only need the world points and camera extrinsics.

Modeling bundle adjustment as an optimisation problem

We use the bundle adjust optimization to solve the SFM problem which boils down to minimizing the re-projection error between the image locations of observed and predicted image points (reprojected points), which is expressed as the sum of squares of a large number of nonlinear, real-valued functions. Thus, the minimization is achieved using nonlinear least-squares algorithms. Of these, Levenberg–Marquardt has proven to be one of the most successful due to its ease of implementation and its use of an effective damping strategy that lends it the ability to converge quickly from a wide range of initial guesses.

Solving the Bundle Adjustment problem

Given a set of features from the previous step, bundle adjustment is applied to simultaneously refine the 3D coordinates, the parameters of the relative motion, and the optical characteristics of the cameras using the reprojection error as a minimization criterion. Bundle adjustment optimizes for both 3D point locations and camera poses.

$$F = \sum_{i=1}^{N_c} \sum_{j=1}^{N_p} \|p_{ij} - \pi(R_i P_j + T_i)\|^2,$$

$$= \sum_{i=1}^{N_c} \sum_{j=1}^{N_p} \left(\frac{e_{ij}^x + f_{ij}^x w_j}{c_{ij} + d_{ij} w_j}\right)^2 + \left(\frac{e_{ij}^y + f_{ij}^y w_j}{c_{ij} + d_{ij} w_j}\right)^2$$

Initialisations for Bundle Adjustment

We know the bundle adjustment will solve for **world points** and **projective matrices** but we must have good initialisation for these values which the algorithm will further optimize. These initialisations used in the paper are as follows:

- The camera intrinsics are already known since in our case the pictures are taken from a known camera.
- The first picture is taken as the reference frame and all camera extrinsics (rotation and translation of the camera to a different view) are with respect to this. Since we are dealing with accidental motions, the translation distances and rotation angles are expected to be very small. Hence, the translations are initialised as a 0 vector and rotation matrices are initialised as identity matrices.
- The 3D world points are initialised with their X and Y coordinates as the reference view's pixel coordinates u and v respectively. The depth for the points is randomly initialised between 2 to 4 meters and the points are parameterised by their inverse depth as this results in a convex optimization problem (explained ahead). So the j^{th} point will be initialised as

$$\mathbf{P}_j = \frac{1}{w_j}[x_j, y_j, 1]^T$$

Convex region for the depths to converge

We derived the final cost function obtained from bundle adjust in terms of the w_is (find link to the pdf below). The proof that this function is convex and the w_is will converge was skipped in the paper, but we went on to derive that as well.

https://drive.google.com/file/d/1iDNOd8-8yhccepE9zSJYgE60Ao64Xpq6/view?usp=sharing

(Added at the end of this PDF)

So after this step of the pipeline, we will have a sparse set of points in the world frame with calculated depths.

DENSE CRF MODEL

Once the sparse 3D structure is obtained from the bundle adjustment we will solve for the CRF model to get the Dense Depth Map.

CRF Model is constructed with the photo-consistency loss for the unary term and Gaussian Kernel in the (spatial and intensity space) as pairwise edge weights.

We will use the Extrinsic parameters obtained from the bundle adjustment to calculate the photo-consistency score.

In common practice CRF models are used for semantic segmentation post processing, here depths are discretized and models as the labelling problem.

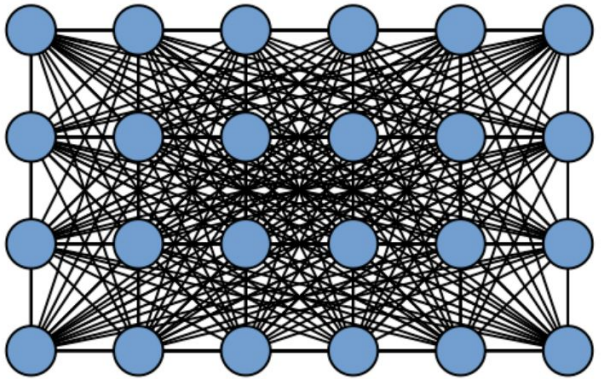


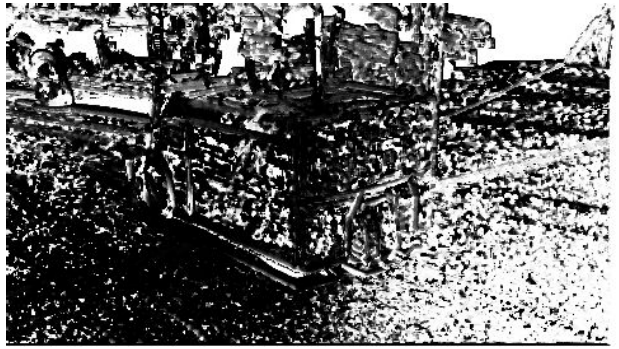
photo credits

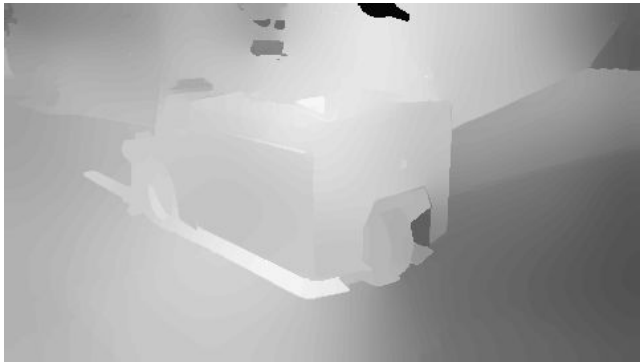
CRF FORMULATION

We will formulate the optimization problem as follows .

$$E(D) = E_p(D) + \alpha E_s(D)$$

Here first term refers to unary (photo consistency) term and second term correspondence to pairwise term . D is the depth map over the optimization happens .



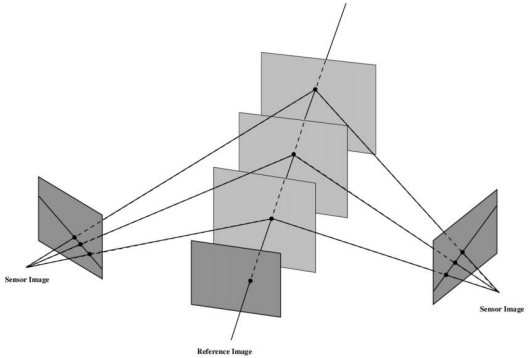


Plane sweep ALgorithm

Reference : Plane sweep paper

The main Idea of the plane sweep algorithm to divide the given space to equal intervals of plane and calculate the photo consistency - score between the reference image (image_0) to every other Image in the sequence .

The reprojection error between the reference image and warped image will be used as unary potential



Now we are have planar world the warping problem will reduce to homography and can be easily deduced .

Homography and unary score

We will calculate the homography between the reference image and every other image using the below formula

$$H_j^{ref}(D) = K * {}^{C}_{W}T_{ref} * D * {}^{C}_{W}T_j^{-1} * K^{-1}$$

The above transformation matrix takes points from the world frame to the jth camera frame and K is the camera intrinsic matrix. For the ith in the jth view we get the following L1 loss:

$$E_p(D) = \sum_j \sum_i |p_{i,ref} - H_j^{ref}(D) * p_{i,j}|$$

We use this loss to get the error between patches of a small window size and averaged across all pairwise Image patches for a particular depth D. In this way, we get a score for every pixel location at different depth values. It represents a probability distribution over depths for that pixel location.

PAIRWISE POTENTIAL

Pairwise cost is modelled as the Gaussian kernel in the position and color space. The algorithm is based on a mean field approximation to the CRF distribution. It has the spatial term which force the depth in the small neighbourhood to take the similar depth labels. It also has an Intensity term which force pixels with similar color will have the consistent depth values.

In the equation below $\rho_c = min(t, |D(i) - D(j)|)$ corresponds to a truncated linear function defined with some threshold t. The pairwise potential is given below.

$$E_s(D) = \sum_{i \in \mathcal{I}, j \in \mathcal{I}, i \neq j} C(i, j, I, L, D) \quad \text{where} \quad C(i, j, I, L, D) = \rho_c(D(i), D(j)) \times \exp\Big(- \underbrace{\frac{||I(i) - I(j)||^2}{\theta_c}}_{\text{Intensity term}} - \underbrace{\frac{||L(i) - L(j)||^2}{\theta_p}}_{\text{Spatial term}} \Big)$$

Mean shift Filtering

Reference : CRF Inference

Algorithm 1 Mean field in fully connected CRFs

Initialize Q
while not converged **do**
$\tilde{Q}_i^{(m)}(l) \leftarrow \sum_{j \neq i} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l)$ for all m
$\hat{Q}_i(x_i) \leftarrow \sum_{l \in \mathcal{L}} \mu^{(m)}(x_i, l) \sum_m w^{(m)} \tilde{Q}_i^{(m)}(l)$
$Q_i(x_i) \leftarrow \exp\{-\psi_u(x_i) - \hat{Q}_i(x_i)\}$
normalize $Q_i(x_i)$
end while

Algorithm 2 Efficient message passing: $\overline{Q}_i^{(m)}(l) = \sum_{j \in \mathcal{V}} k^{(m)}(\mathbf{f}_i, \mathbf{f}_j) Q_j(l)$

$Q_{\downarrow}(l) \leftarrow \text{downsample}(Q(l))$ ▷ **Downsample**
$\forall_{i \in \mathcal{V}_{\downarrow}} \overline{Q}_{\downarrow i}^{(m)}(l) \leftarrow \sum_{j \in \mathcal{V}_{\downarrow}} k^{(m)}(\mathbf{f}_{\downarrow i}, \mathbf{f}_{\downarrow j}) Q_{\downarrow j}(l)$ ▷ **Convolution** on samples $\mathbf{f}_{\downarrow}$
$\overline{Q}^{(m)}(l) \leftarrow \text{upsample}(\overline{Q}_{\downarrow}^{(m)}(l))$ ▷ **Upsample**

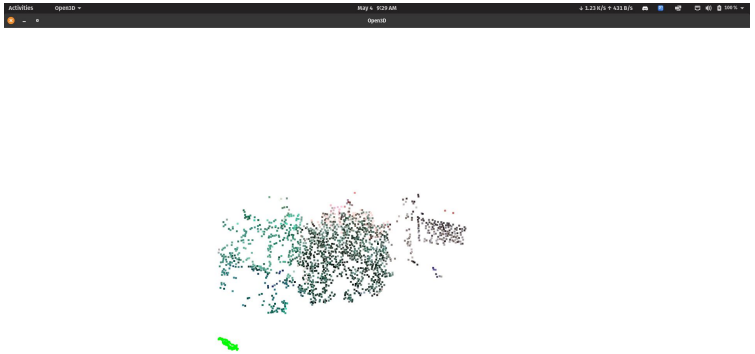
Results

Our code takes a **sequence of images** from a video and, for output, generates the images for **optical flow, sparse point clouds before and after the bundle adjustment, cost volume, sparse and dense depth maps**.

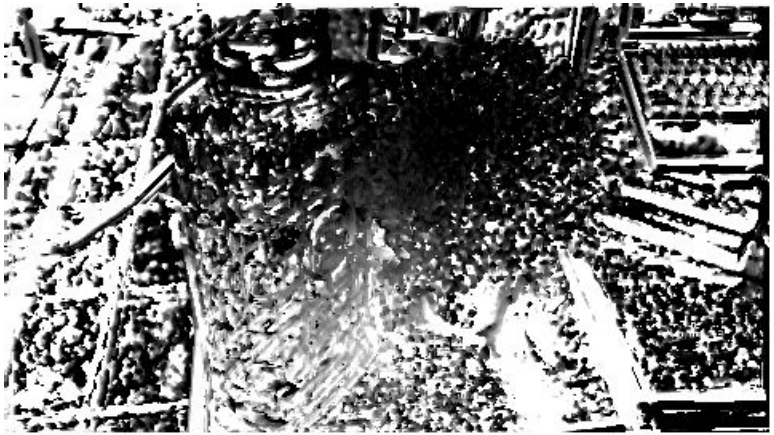
Next few slides obtain the output images for optical flow, sparse point cloud after the bundle adjustment, cost volume and dense depth map graph for visualization.

We'll also show present the output generation live.

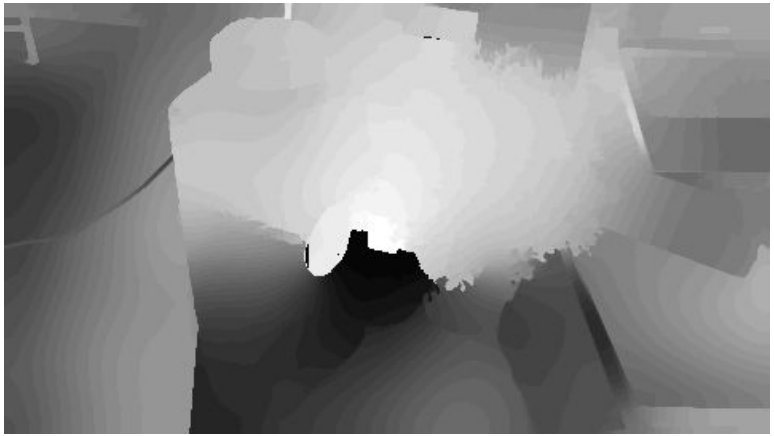
Optical flow



Sparse Point Cloud

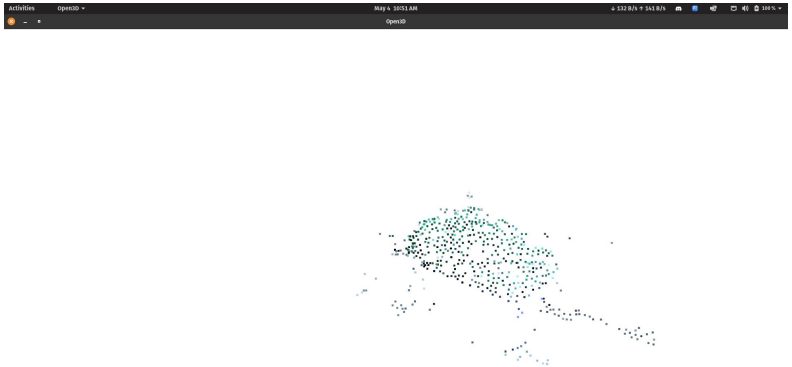


Crude Depth



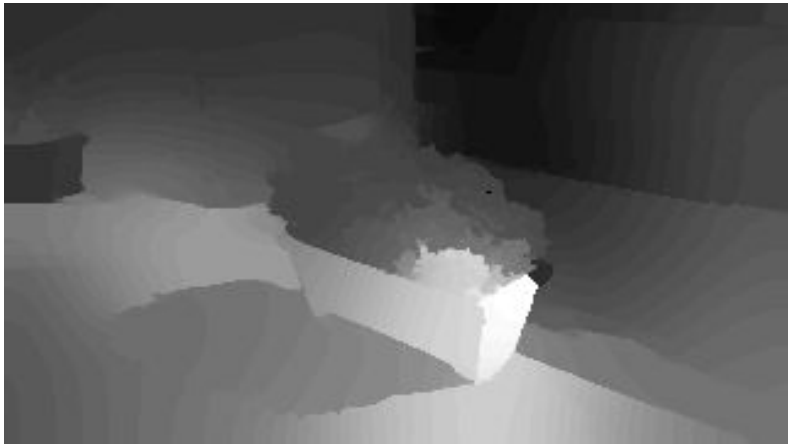
Depth Map

Optical flow



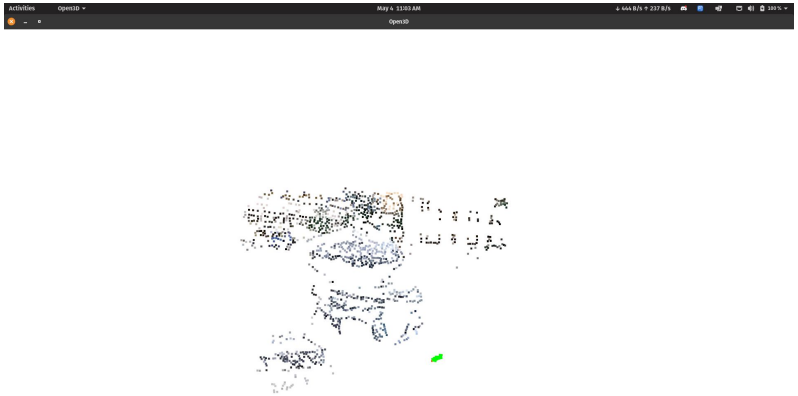
Sparse Point Cloud

Crude Depth



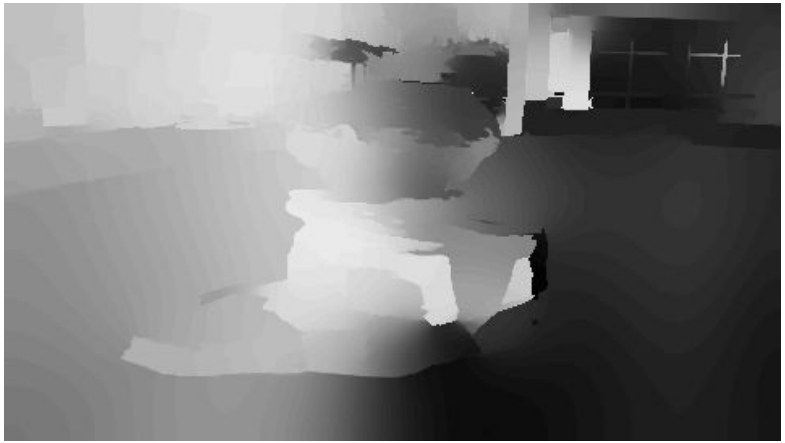
Depth Map

Optical flow



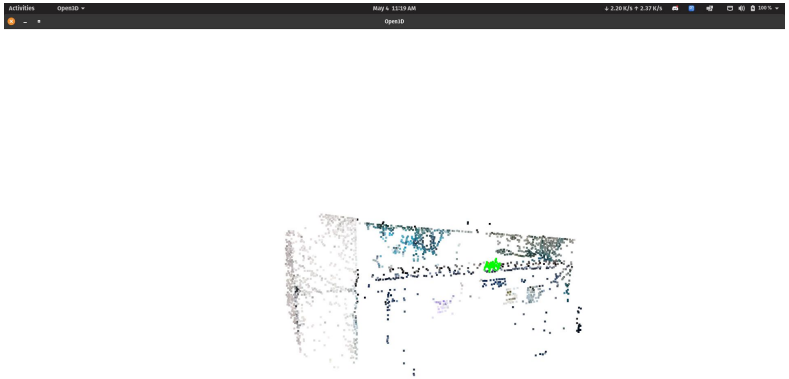
Sparse Point Cloud

Crude Depth

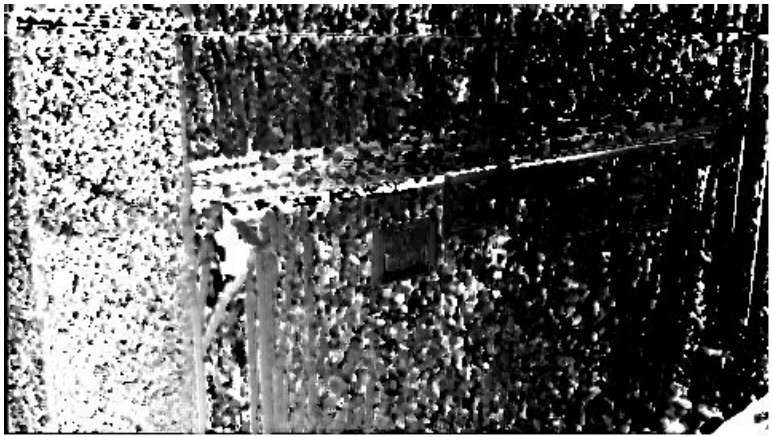


Depth Map

Optical flow



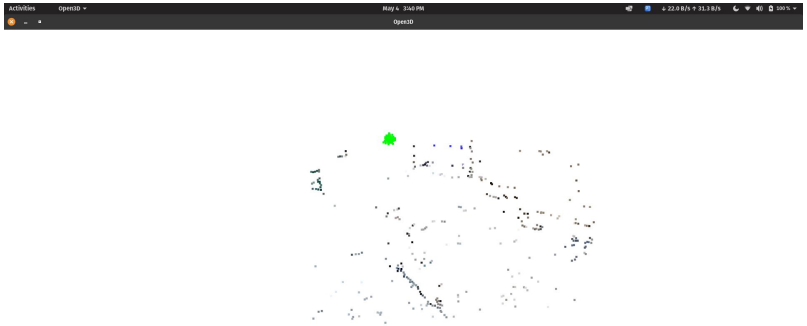
Sparse Point Cloud



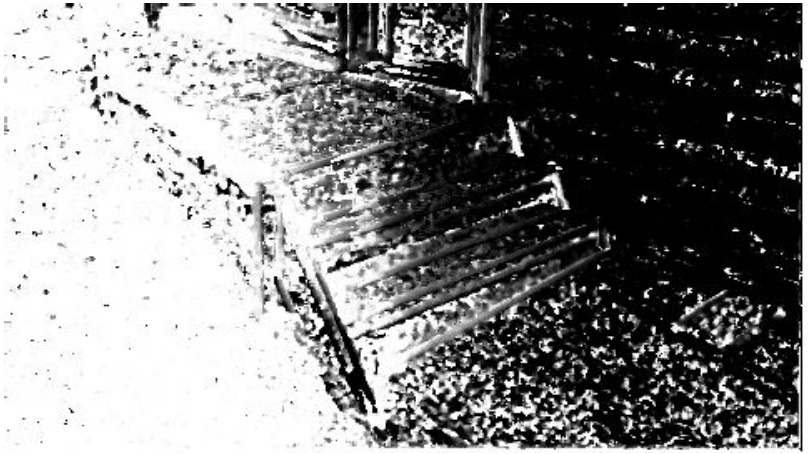
Crude Depth

Depth Map

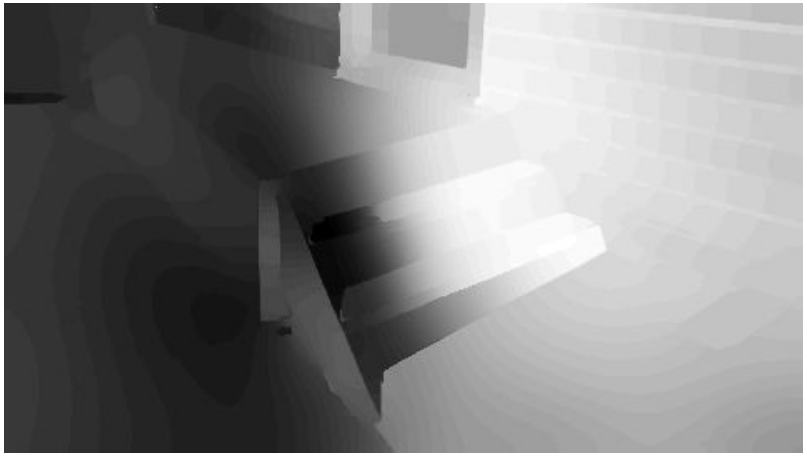
Optical flow



Sparse Point Cloud



Crude Depth



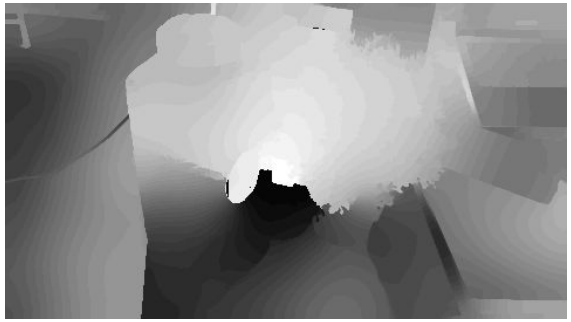
Depth Map

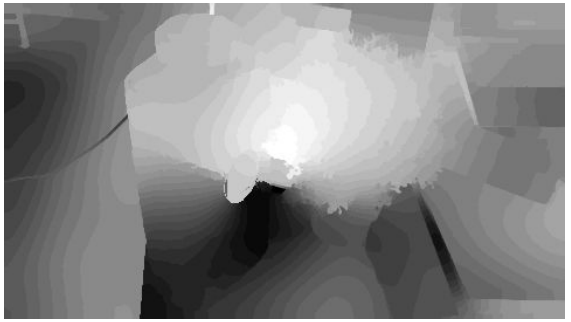
Experimenting with Parameters

Maximum penalty for DenseCRF

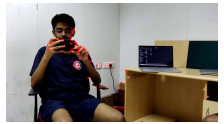
The pairwise edge weights are proportional to ρ_c which is equal to the minimum of t (i.e., the max penalty) and absolute value of *D(i) - D(j)*. We see that if the value of maximum penalty is small then the pairwise edge weights can’t be large and hence final segmentation would rely more on Unary weights, which would make the image less continuous. This is reflected in the results.

Maximum penalty for DenseCRF





t=0.1 t=0.25 t=0.35



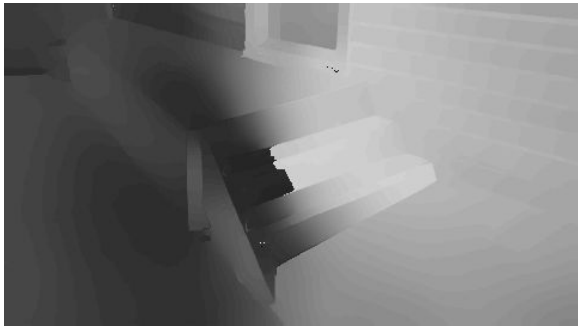
t=0.1 t=0.25 t=0.35

Varying weight for DenseCRF

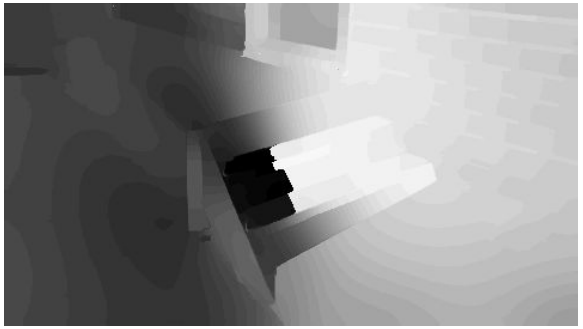
$$E(D) = E_p(D) + \alpha E_s(D)$$

It determines how much significance we are giving to the pairwise potential. If alpha is high, then image should be more continuous and must rely less on unary weights.

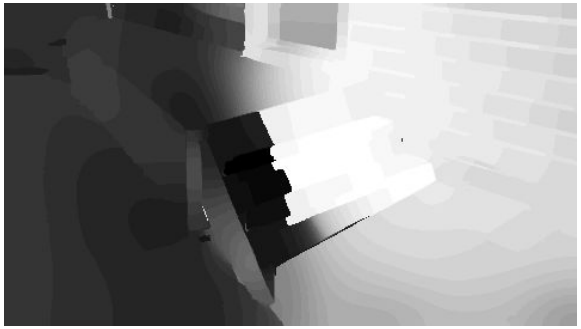
Varying weight for DenseCRF



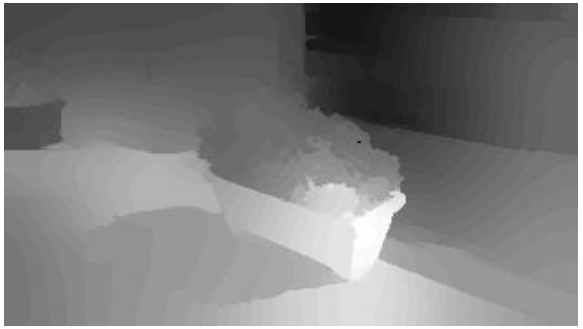
W = 0.5



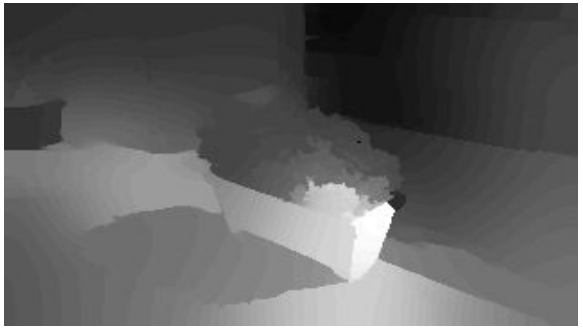
W = 1.0



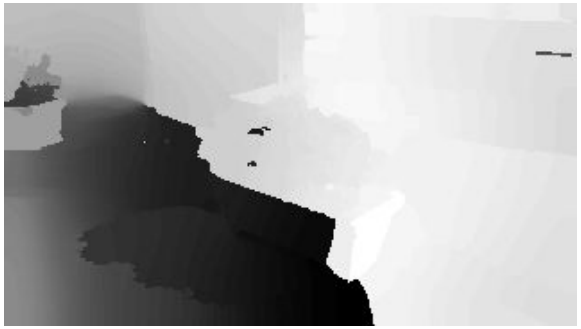
W = 2.5



W = 0.5



W = 1.0



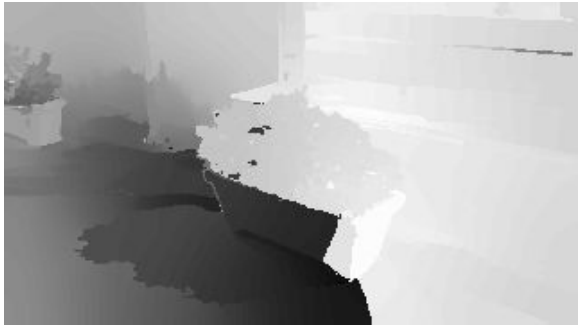
W = 2.5

Intensity standard deviation for pairwise potentials

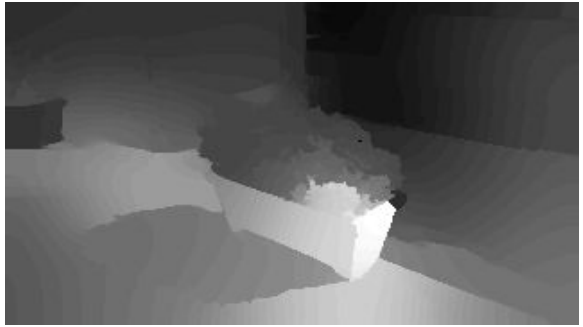
The pairwise edge weights are proportional to $\exp\left(-\underbrace{\frac{||I(i)-I(j)||^2}{\theta_c}}_{\text{Intensity term}}-\underbrace{\frac{||L(i)-L(j)||^2}{\theta_p}}_{\text{Spatial term}}\right)$

By varying θ_c we can change the dependence on the intensity part of the pairwise edge weight term. If θ_c is sufficiently small then we will be giving very high penalty for slight difference between the intensity values, and comparatively lower penalty for spatial difference. Therefore, pixels located nearby will have a chance having the same label.

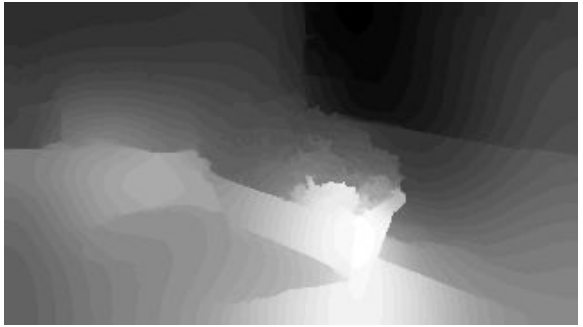
Intensity standard deviation for pairwise potentials



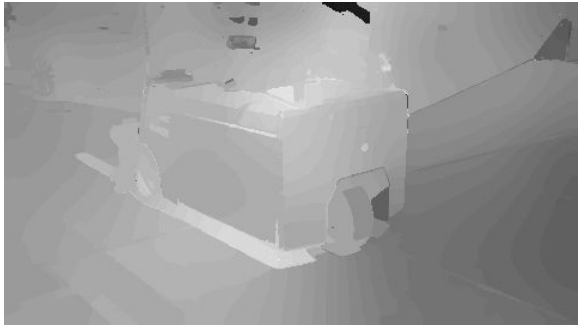
$\theta_c = 10$



$\theta_c = 20$



$\theta_c = 30$



$\theta_c = 10$

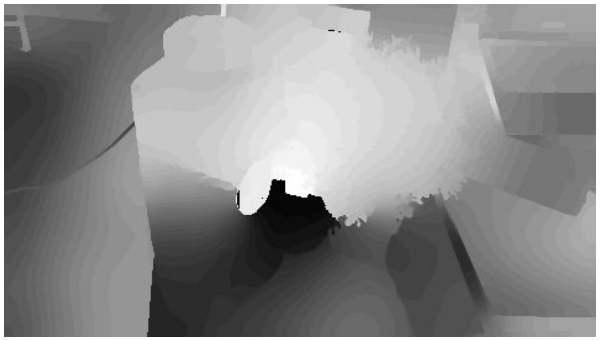
$\theta_c = 20$

$\theta_c = 30$

Patch Radius for calculating unary potential

In our experiments 3x3 windows is producing good results. From the observations increasing the window size will results in loss of local features and making it global . this will results in loss of finer grained details on the depth map.

Patch Radius for calculating unary potential



R=1

R=2

R=3

R=1

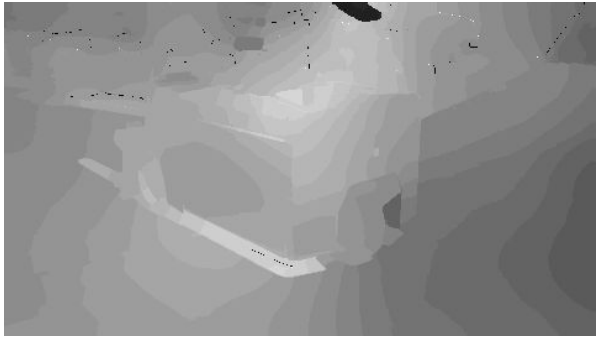
R=2

R=3

Number of depth samples for CRF

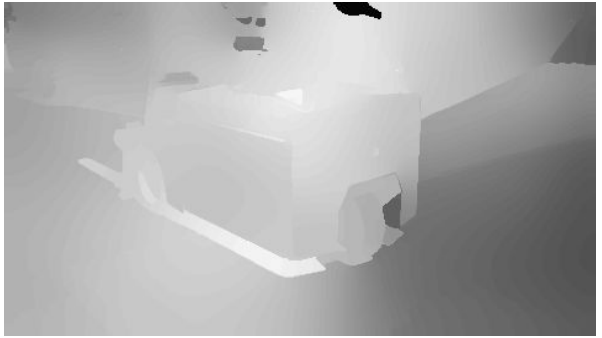
The output of CRF is labels for each pixel of the reference frame. Labels depict the depth of each pixel. Due to our assumption that each point in the image lies between 2 to 4 m from the camera, we have discretized the distance and the number of possible depth samples (i.e., distances from the camera center) are varied. Therefore, if the number of depth samples are more, the image would be more continuous.

Number of depth samples for CRF

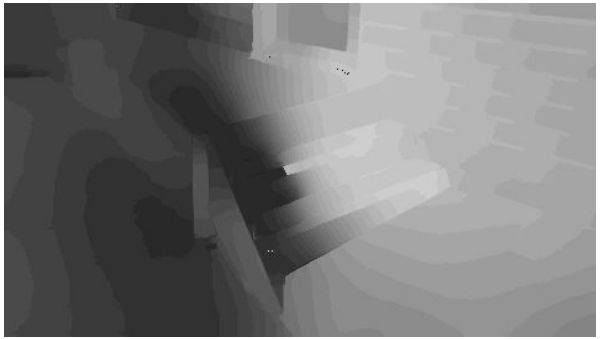


32 Planes

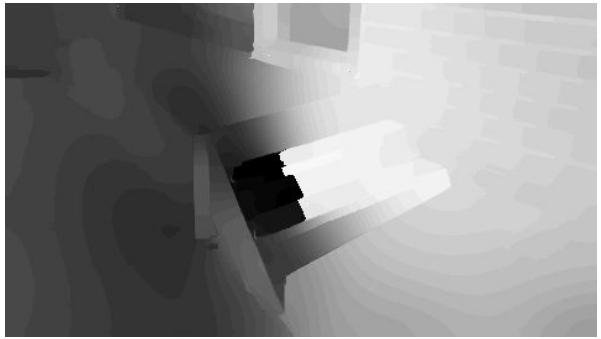
64 Planes



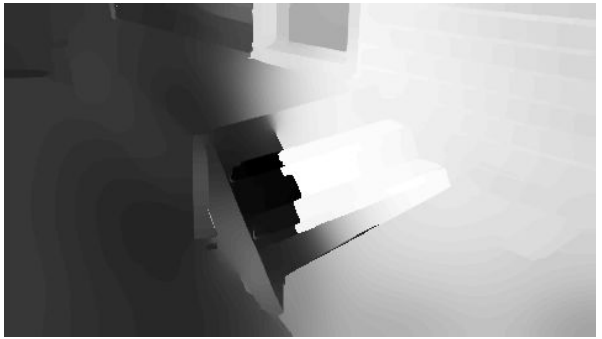
128 Planes



32 Planes



64 Planes



128 Planes

Questions?

CV : TeamX (145)

Bundle Adjustment Optimization
in 3D reconstruction from accidental motion

Aim - Showing that there is a range of w_i s for which the bundle adjustment cost function will be convex.

Working -

- The i^{th} point is : $P_i = \frac{1}{w_i} [x_i \; y_i \; 1]$

- The world ~~point~~ to camera rotation matrix for j^{th} camera (using x-y-z Euler angles and assuming $\theta_j^x, \theta_j^y, \theta_j^z \approx 0$)

$$ {}^{c}_{w}R_j = \begin{bmatrix} 1 & -\theta_j^z & \theta_j^y \\ \theta_j^z & 1 & -\theta_j^x \\ -\theta_j^y & \theta_j^x & 1 \end{bmatrix} $$

- Similarly, translation matrix:

$$ {}^{c}_{w}T_j = \begin{bmatrix} T_j^x \\ T_j^y \\ T_j^z \end{bmatrix} $$

- Projecting point P_i to j^{th} camera

$$ {}^{c}_{w}R_j P_i + {}^{c}_{w}T_j = \begin{bmatrix} 1 & -\theta_j^z & \theta_j^y \\ \theta_j^z & 1 & -\theta_j^x \\ -\theta_j^y & \theta_j^x & 1 \end{bmatrix} \begin{bmatrix} x_i/w_i \\ y_i/w_i \\ 1/w_i \end{bmatrix} + \begin{bmatrix} T_j^x \\ T_j^y \\ T_j^z \end{bmatrix} $$

$$ = \frac{1}{w_i} \begin{bmatrix} a_{ij}^x \\ a_{ij}^y \\ c_{ij} \end{bmatrix} + \begin{bmatrix} b_{ij}^x \\ b_{ij}^y \\ d_{ij} \end{bmatrix} $$

To optimize for w_i, we do -

$$\underset{w_i}{\text{argmin}} \left(p_{ij}^x - \frac{a_x^{ij}}{\frac{c_{ij}}{w_i} + d_{ij}} + b_{ij}^x \right)^2 + \left(p_{ij}^y - \frac{\frac{a_y^{ij}}{w_i} + b_{ij}^y}{\frac{c_{ij}}{w_i} + d_{ij}} \right)^2$$

$$\Rightarrow \underset{w_i}{\text{argmin}} \left(\frac{(p_{ij}^x c_{ij} - a_{ij}^x) + (p_{ij}^x d_{ij} - b_{ij}^x) w_i}{c_{ij} + d_{ij} w_i} \right)^2$$

$$+ \left(\frac{(p_{ij}^y c_{ij} - a_{ij}^y) + (p_{ij}^y d_{ij} - b_{ij}) w_i}{c_{ij} + d_{ij} w_i} \right)^2$$

Rewrite as:

$$\underset{w_i}{\text{argmin}} \left(\frac{e_{ij}^x + f_{ij}^x w_i}{c_{ij} + d_{ij} w_i} \right)^2 + \left(\frac{e_{ij}^y + f_{ij}^y w_i}{c_{ij} + d_{ij} w_i} \right)^2$$

$$\Rightarrow \underset{w_i}{\text{argmin}} \left(\frac{f_{ij}^x}{d_{ij}} \right) \left(\frac{e_{ij}^x / f_{ij}^x + w_i}{c_{ij} / d_{ij} + w_i} \right)^2 + \left(\frac{f_{ij}^y}{d_{ij}} \right) \left(\frac{e_{ij}^y / f_{ij}^y + w_i}{c_{ij} / d_{ij} + w_i} \right)^2$$

These two terms are of the form $\left(\frac{x-a}{x-b} \right)^2$ where $a > b$

since $-e_{ij}^x / f_{ij}^x > -c_{ij}/d_{ij}$ (since $c_{ij} \approx 1$ and $d_{ij} \approx 0$).

$\left(\frac{x-a}{x-b} \right)^2$ when $a > b$ is known to be convex in the range $\left(b, \frac{3a}{2} - \frac{b}{2} \right)$.

Taking all approximations and using the standard result, we get $w_i \in \left(0, \left| \frac{c_{ij}}{2 d_{ij}} \right| \right)$. We choose the j which gives the minimum $|c_{ij}/2d_{ij}|$ to get the convex region for w_i.