(a) Input image sequence (b) Foreground points of SfM (c) Dense reconstruction

3D Reconstruction from accidental motion

Team: SfM

Link to the Project Code

Aim & Problem Formulation

Aim

3D scene reconstruction from a set of initial frames of a video capture by exploiting *accidental motion*. *Accidental motion* is the small jerky movements that occur when trying to hold an imaging device still.

Input: Sequence of frames part of a video

Output: Dense depth map of a scene from a reference view

Problem formulation

- Given an image sequence of N_C images, we apply **KLT tracking** to track a set of N_P points seen from every image. We assume each image was captured by a different camera with unique extrinsic parameters.
- **Bundle Adjustment** is used to estimate the ground-truth 3D world coordinates along with the extrinsic parameters of every camera using the initially tracked points.
- Using the estimated camera poses, the 3D scene is densely reconstructed as a single smooth depth map. A **Conditional Random Field** is used to minimize an energy function using plane-sweep approach and mean-field.

Recap

Step 1: KLT Tracking

We first track features between all the image frames using KLT tracking method. The algorithm is described as below.

- Detect Shi-Tomasi features in the reference image using the scoring function:

$$R = min(\lambda_1, \lambda_2)$$

- Estimate Lucas-Kanade optical flow over all the images in the sequence to find out good trackable features across all images.
- Filter out corners by estimating the homography between the reference frame and every other frame in the image sequence.
- Select those corners that are inliers for more than 95 % of images found by estimating homography.

Note: The first image is considered as reference frame for the image sequence.

Step 2: Structure from motion

- 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.
- Reprojection error is the error between the projected 3D point on the image frame and the observed pixel.
- Bundle adjustment optimizes for both 3D point locations and camera poses.

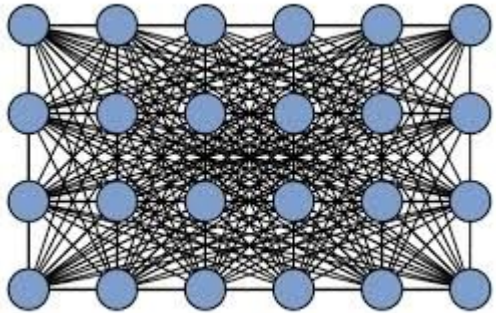
$$F = \sum_{i=1}^{N_c} \sum_{j=1}^{N_p} \|p_{ij} - \pi(R_i P_j + T_i)\|^2,$$

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

Note: The intrinsic parameters are same for all cameras. All cameras (except the reference camera) are initialized with zero rotation and translation. 3D points are initialized by its inverse depth relative to an arbitrary reference frame. All the camera poses have random depth initialization.

Step 3: Dense reconstruction

Given an initial estimate of the 3D scene from the previous step, we construct a dense depth map of the scene from the reference view. We use the extrinsic parameters of all the cameras to compute a photo-consistency score. A Dense CRF model is constructed with the photo-consistency term as the unary potentials and use a Gaussian kernel in an arbitrary feature space (spatial & intensity) as pairwise edge potentials.



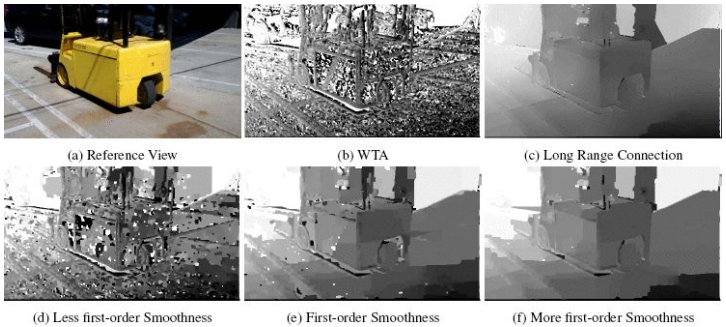
In a Dense CRF, each node is connected to every other node. Each pixel here is a node of the network.

Conditional Random Fields

- We use a CRF formulation to minimize the below energy function.

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

- Here, $E_p(D)$ is the unary (photo-consistency) term and $E_s(D)$ is the pairwise term. D is the dense depth map that is optimized over.
- We formulate this problem of obtaining a dense depth map as a labelling problem where the depth values are the labels and each pixel is a node. The unary potential of every node gives an initial estimate of the depth locations.

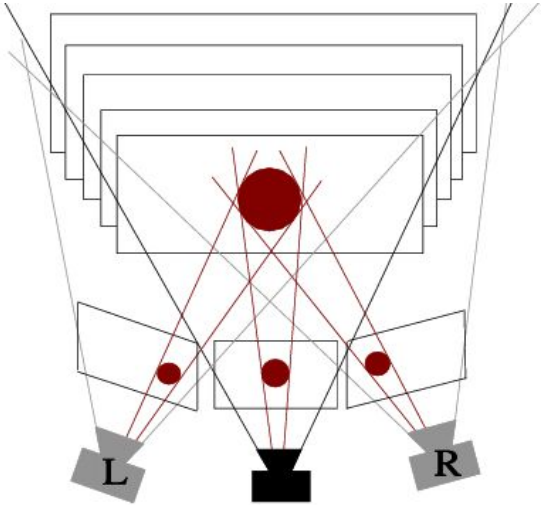


(a) Reference View (b) WTA (c) Long Range Connection

(d) Less first-order Smoothness (e) First-order Smoothness (f) More first-order Smoothness

Unary potential - Plane sweep method

The plane-sweeping method is used to obtain a photo-consistency score of the 3D scene given the image sequence and camera parameters. It sweeps a plane varying the z-axis (different depths) to calculate the photo-consistency score. We use this to initialize the nodes with unary potentials.



The fully-connected CRF allows pixel connections with longer range so that the photo-consistency measurement can be effectively aggregated from an area to a pixel in it.

Photo-consistency score

We first sample different depth values in the acceptable depth range of the 3D scene. To calculate the unary potentials we compute a homography between the reference image and every other image. The homography matrix $H_j^{ref}(D)$ between the reference image and jth camera frame at these different depths D is computed as follows:

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

Here, ${}^{C}_{W}T_j$ is the transformation matrix of the world with respect to the jth camera frame and K is the camera intrinsic matrix. For the ith in the jth view $p_{i,j}$ we get the following L1 loss:

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

The above L1 loss is computed between patches of a small window size and averaged across all pairwise 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.

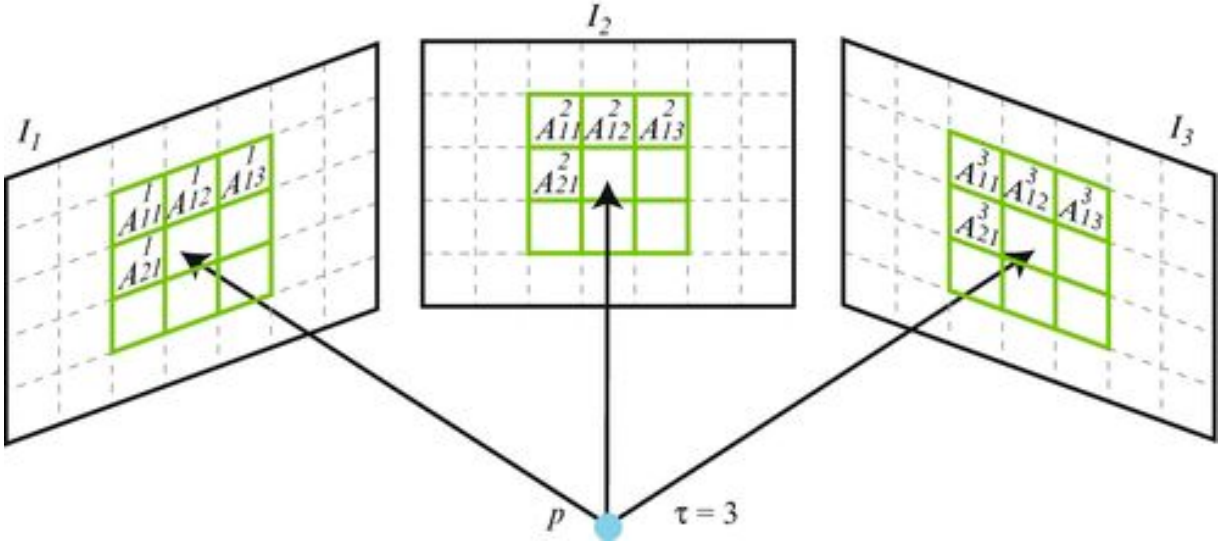


Photo-consistency of a 3D point in three different views. Small 3x3 patch considered.

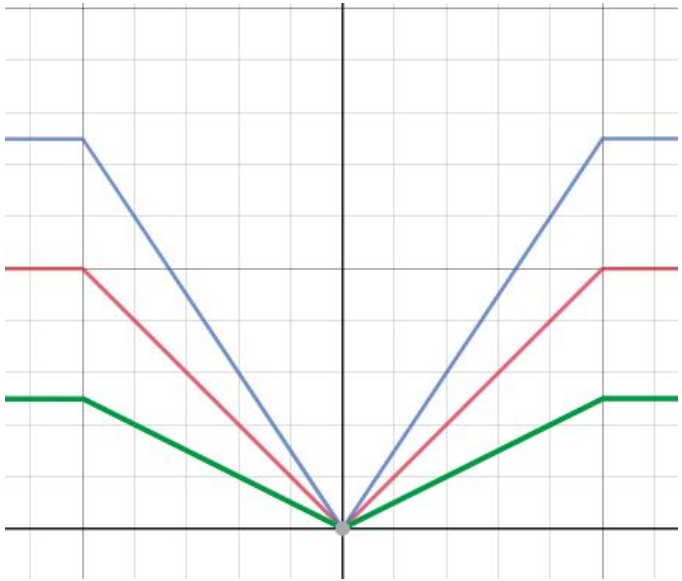
Pairwise potential - Gaussian kernel

The pairwise potential is a Gaussian kernel in arbitrary feature space.

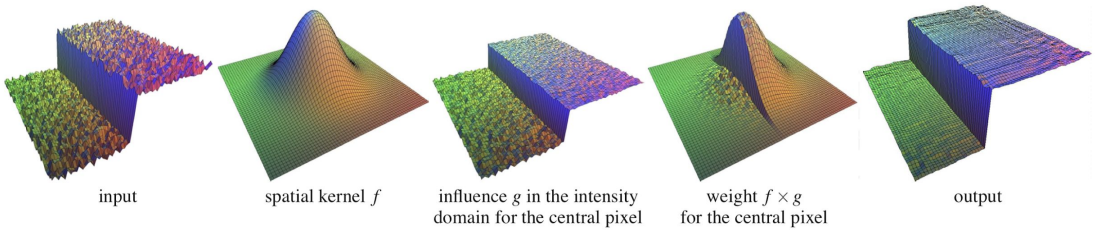
- It has a spatial term such that the depth within a small neighbourhood is consistent
- It has an intensity term such that pixels within an area with similar colours have consistent depth, since they are likely to belong to the same object
- The pairwise potential is formulated as below.
  - $E_s(D) = \sum_{i \in \mathcal{I}, j \in \mathcal{I}, i \neq j} C(i,j,I,L,D)$ where $C(i,j,I,L,D) = \rho_c(D(i),D(j)) \times \exp\Big(-\underbrace{\frac{||I(i)-I(j)||^2}{\theta_c}}_{\text{Intensity term}} - \underbrace{\frac{||L(i)-L(j)||^2}{\theta_p}}_{\text{Spatial term}} \Big)$

In the above equation $\rho_c = min(t, |D(i) - D(j)|)$ is the truncated linear function defined with some threshold *t*.

We also weigh the truncated linear function with a weight *w*. The effect of *t* and *w* have been discussed.



Truncated linear function with different weights



The Gaussian kernel term is similar to the bilateral filter

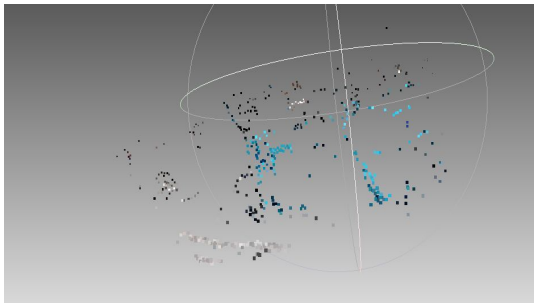
Results

Results

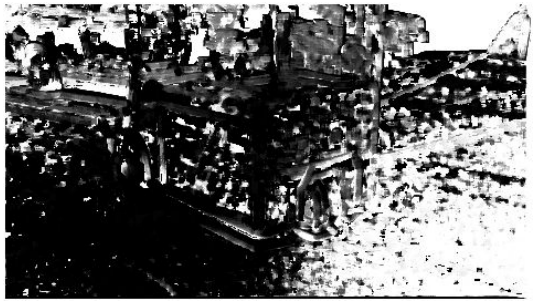
Reference Image

Optical Flow

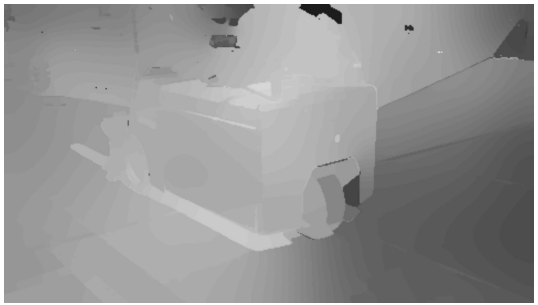
Sparse Depth Map



Sparse Point Cloud



WTA



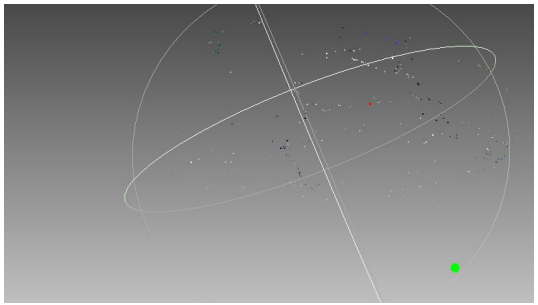
Dense Depth Map

Results

Reference Image

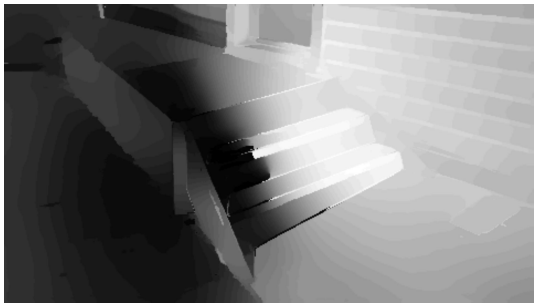
Optical Flow

Sparse Depth Map



Sparse Point Cloud

WTA

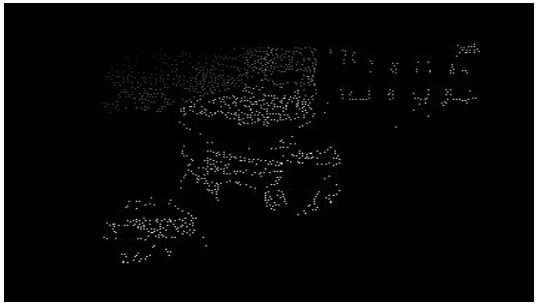


Dense Depth Map

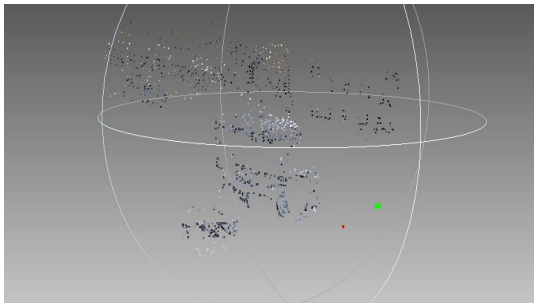
Results

Reference Image

Optical Flow

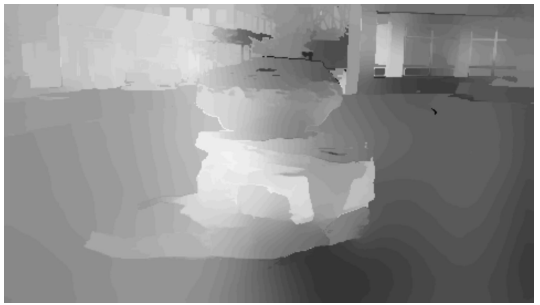


Sparse Depth Map



Sparse Point Cloud

WTA



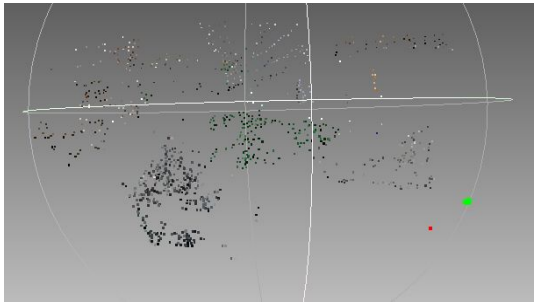
Dense Depth Map

Results

Reference Image

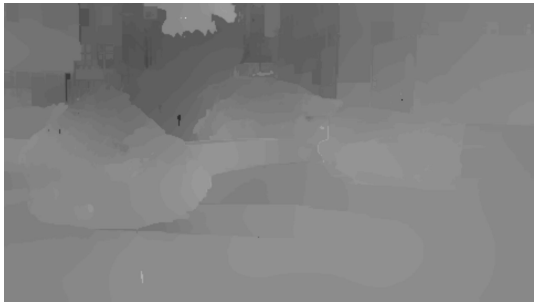
Optical Flow

Sparse Depth Map



Sparse Point Cloud

WTA

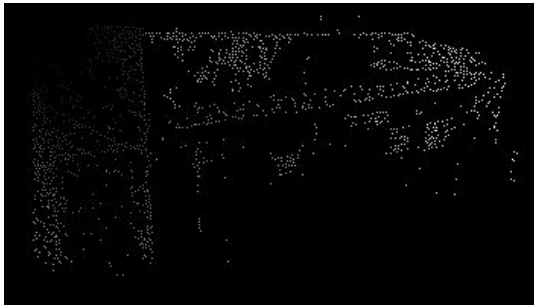


Dense Depth Map

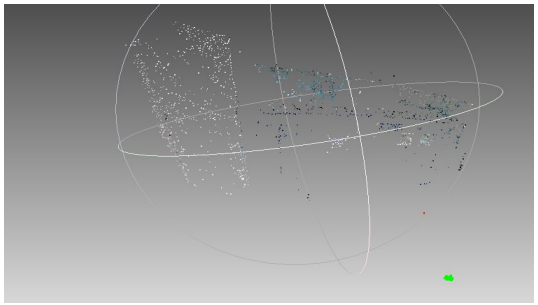
Results

Reference Image

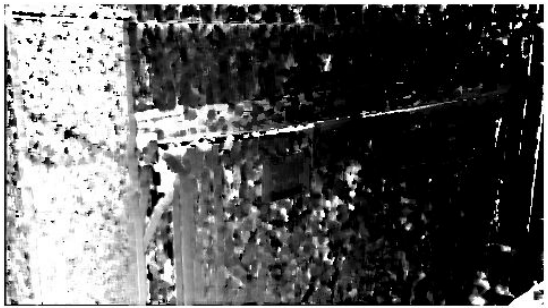
Optical Flow



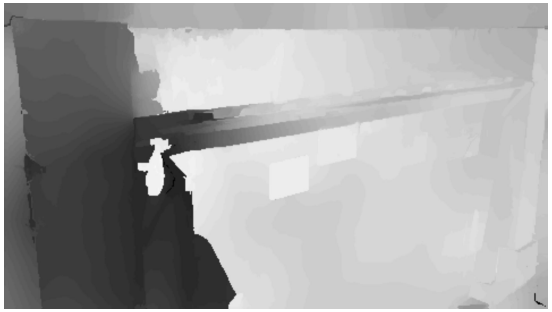
Sparse Depth Map



Sparse Point Cloud



WTA



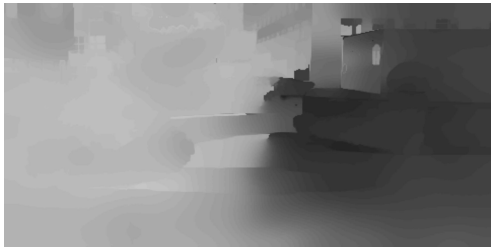
Dense Depth Map

Experiments

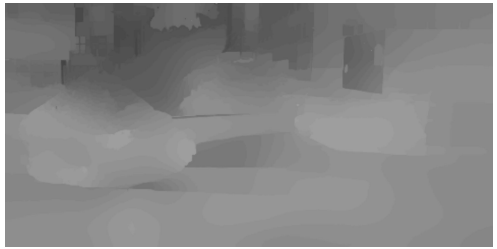
Experiment 1

Effect of multiple images

Results



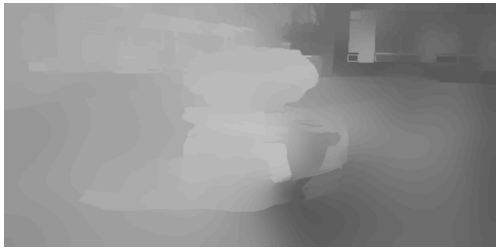
30 Frames



50 Frames

100 Frames

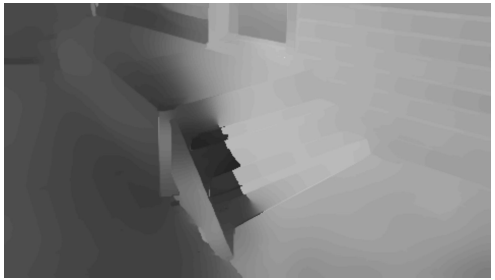
30 Frames



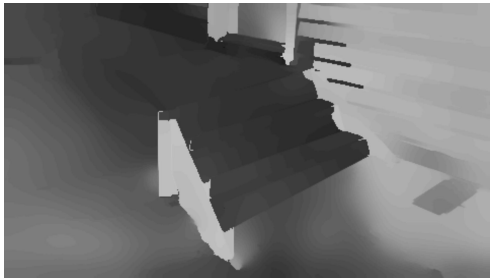
50 Frames

100 Frames

Results



30 Frames



50 Frames

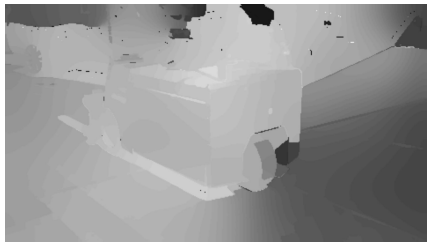
100 Frames

Experiment 2

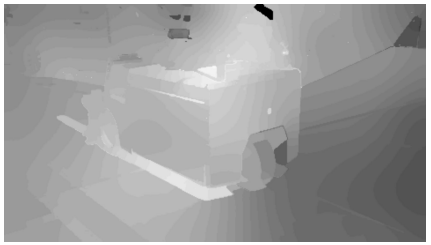
Effect of t in $\rho_c(.)$

Results

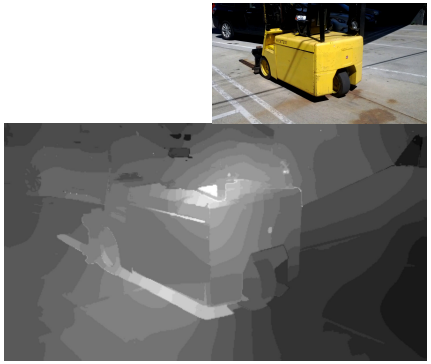
0.05



0.15

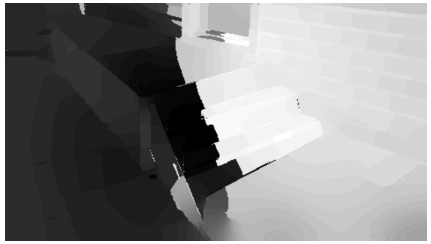


0.25

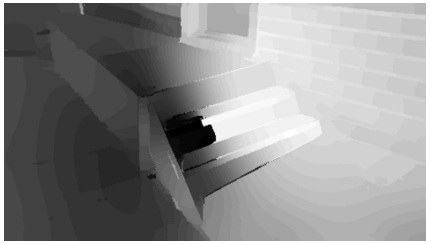


0.40

0.05



0.15

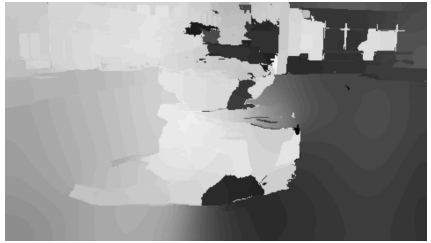


0.25

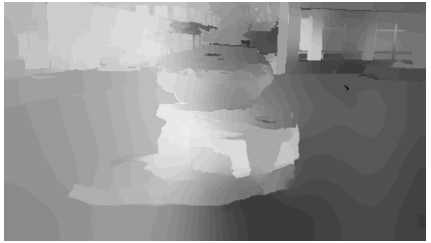
0.40

Results

0.05



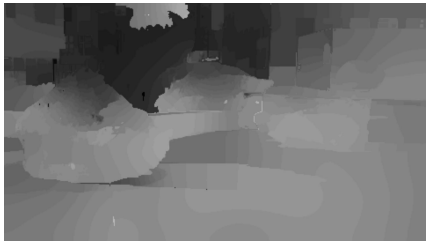
0.15



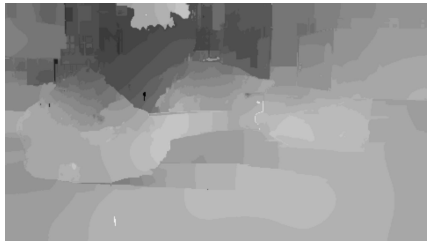
0.25

0.40

0.05



0.15



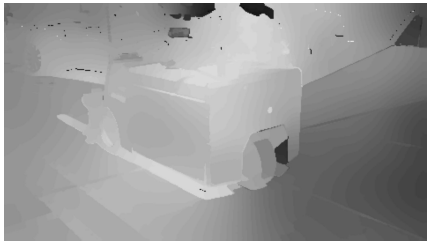
0.25

0.40

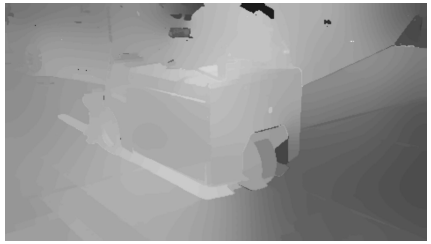
Experiment 3

Effect of w in $\rho_c(.)$

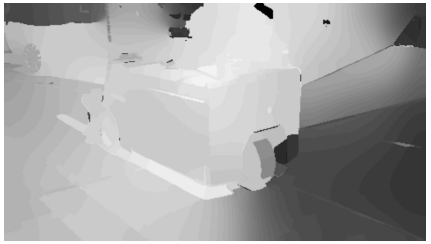
Results



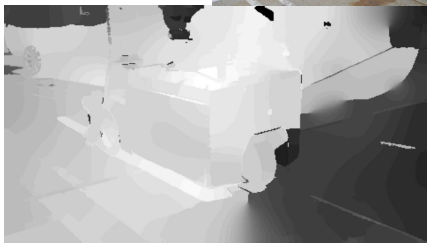
0.5



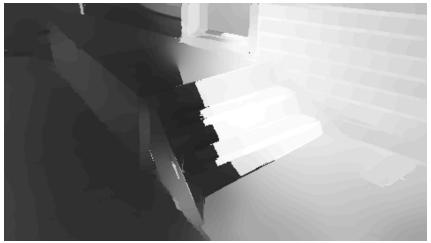
1.0



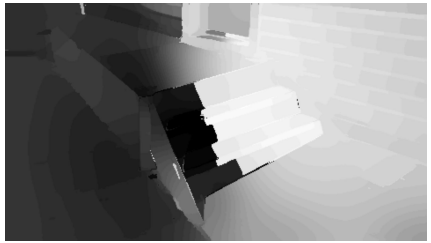
2.0



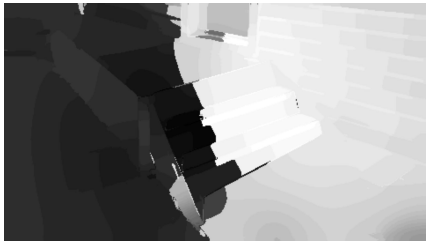
3.0



0.5



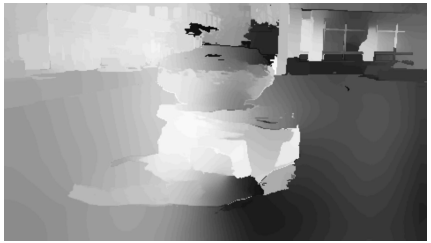
1.0

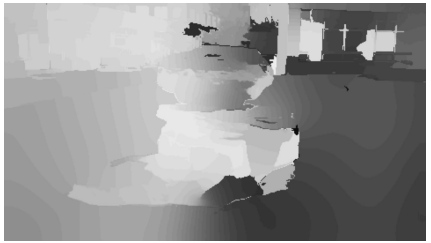


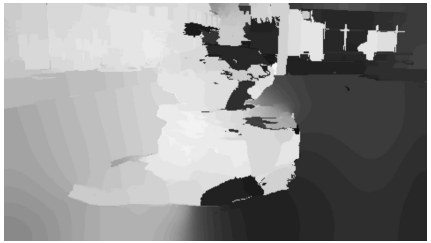
2.0

3.0

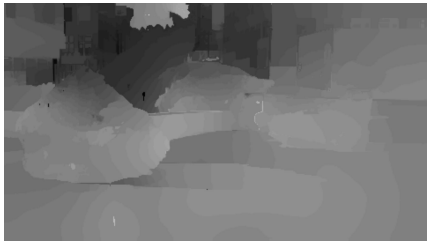
Results

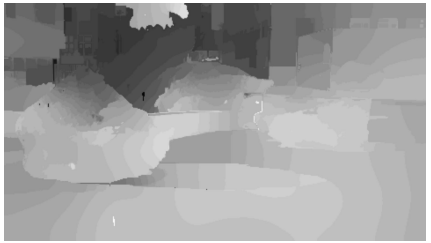


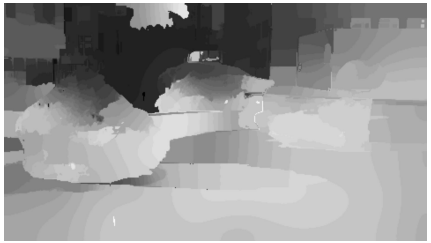


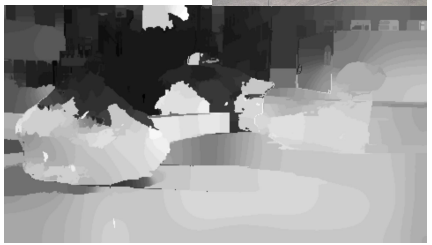


0.5 1.0 2.0 3.0







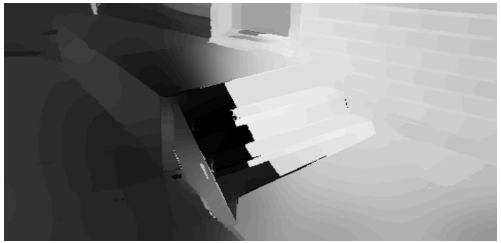


0.5 1.0 2.0 3.0

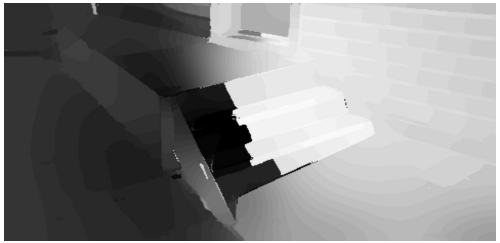
Experiment 4

Effect of patch size on E_p

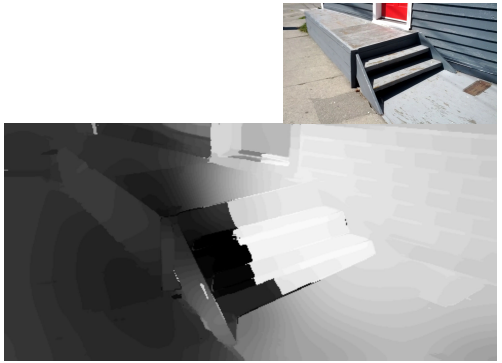
Results



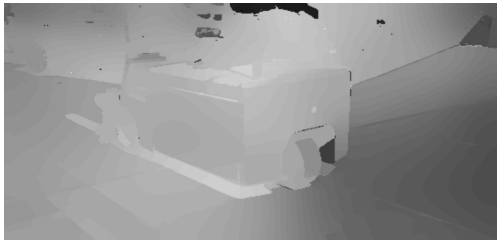
Size=3



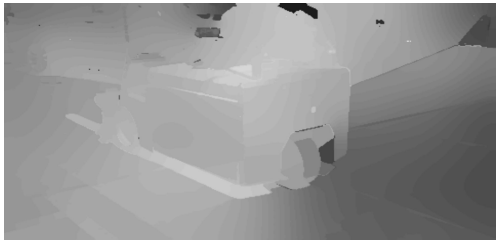
Size=5



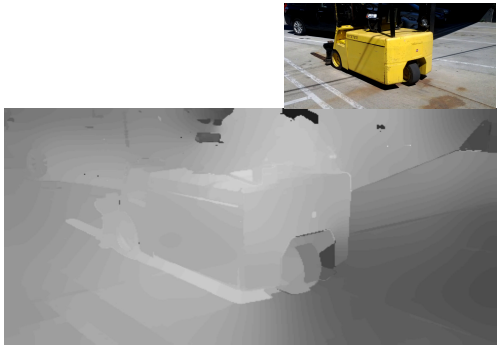
Size=7



Size=3

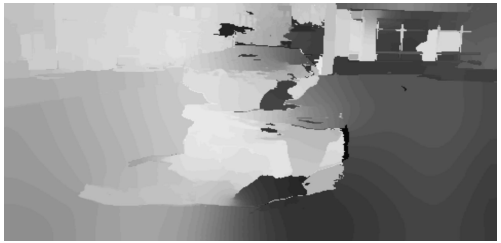


Size=5

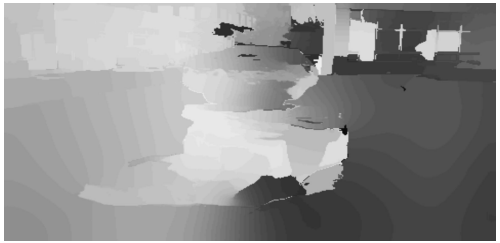


Size=7

Results

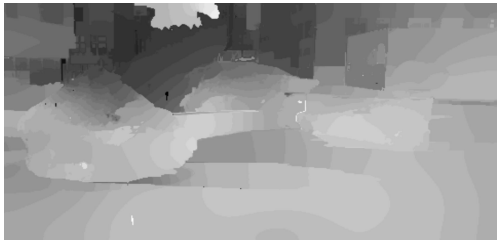


Size=3

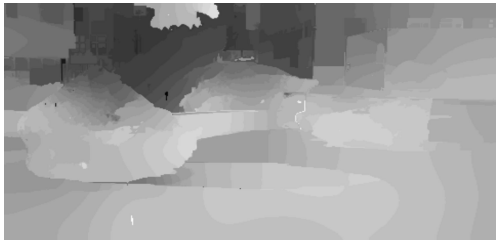


Size=5

Size=7



Size=3



Size=5

Size=7

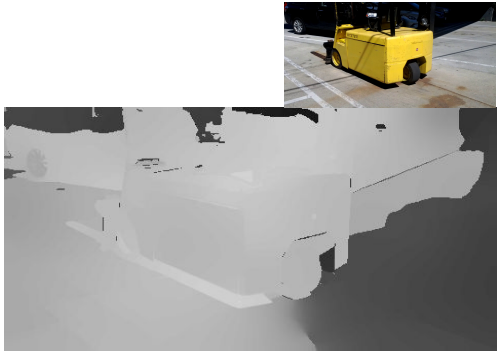
Experiment 5

Effect of multiple depth samples

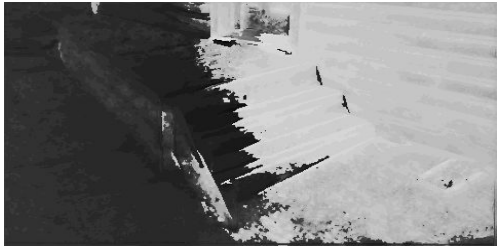
Results

32 Samples

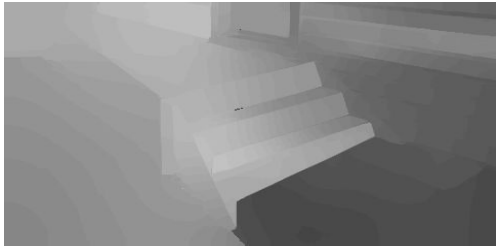
64 Samples



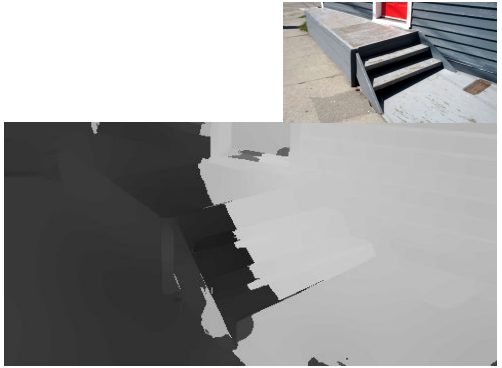
128 Samples



32 Samples



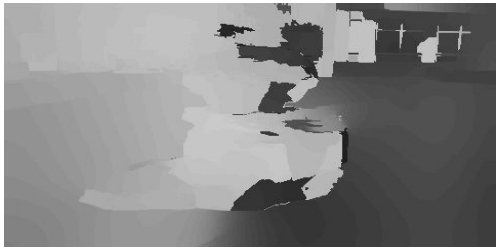
64 Samples



128 Samples

Results

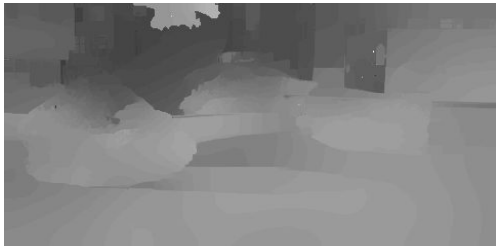
32 Samples



64 Samples

128 Samples

32 Samples



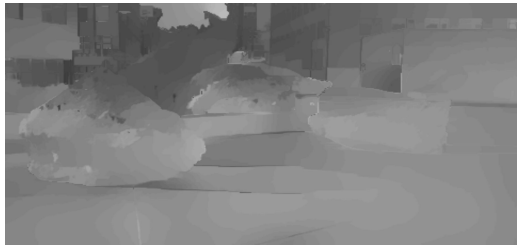
64 Samples

128 Samples

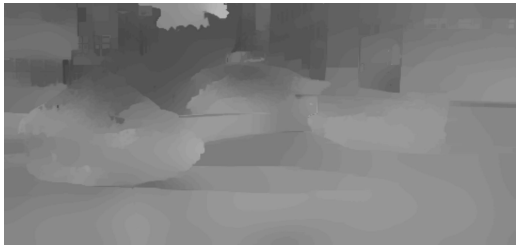
Experiment 6

Effect of θ_c

Results

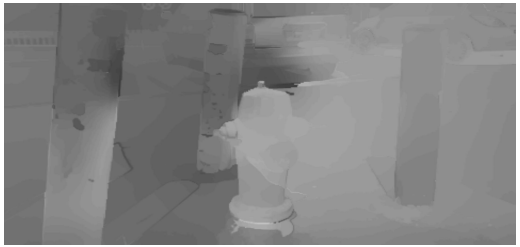


θ_c = 10

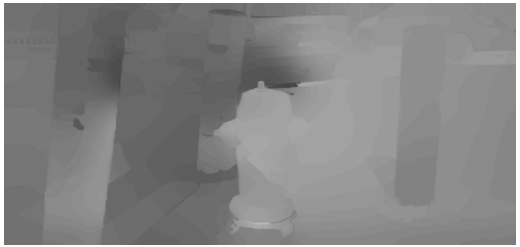


θ_c = 20

θ_c = 35



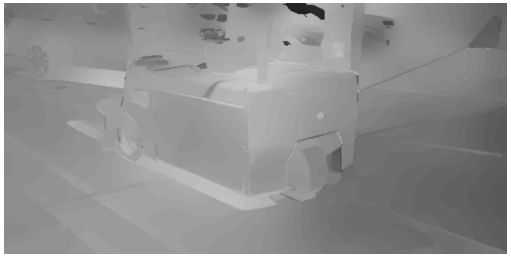
θ_c = 10



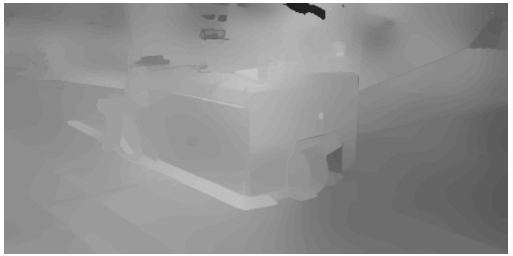
θ_c = 20

θ_c = 35

Results

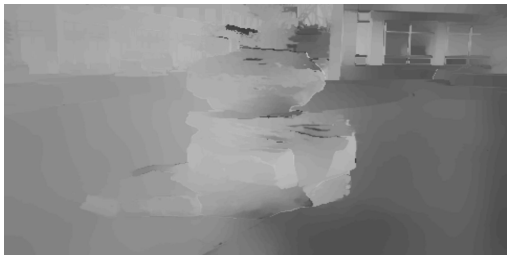


θ_c = 10

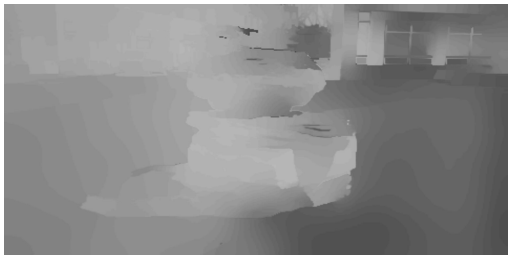


θ_c = 20

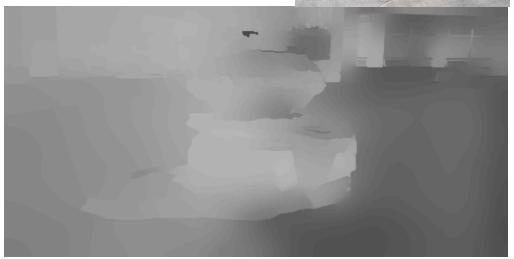
θ_c = 35



θ_c = 10



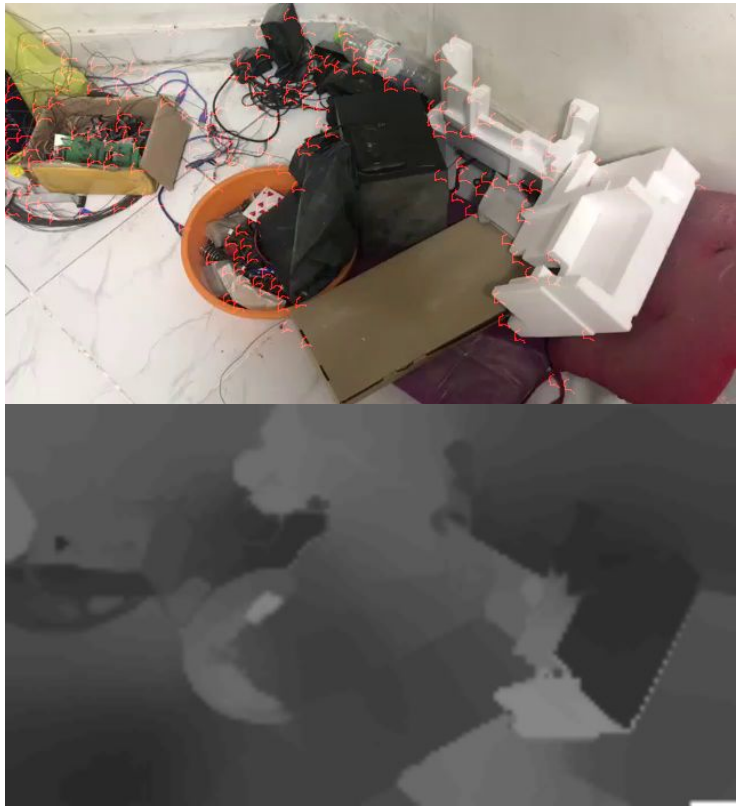
θ_c = 20



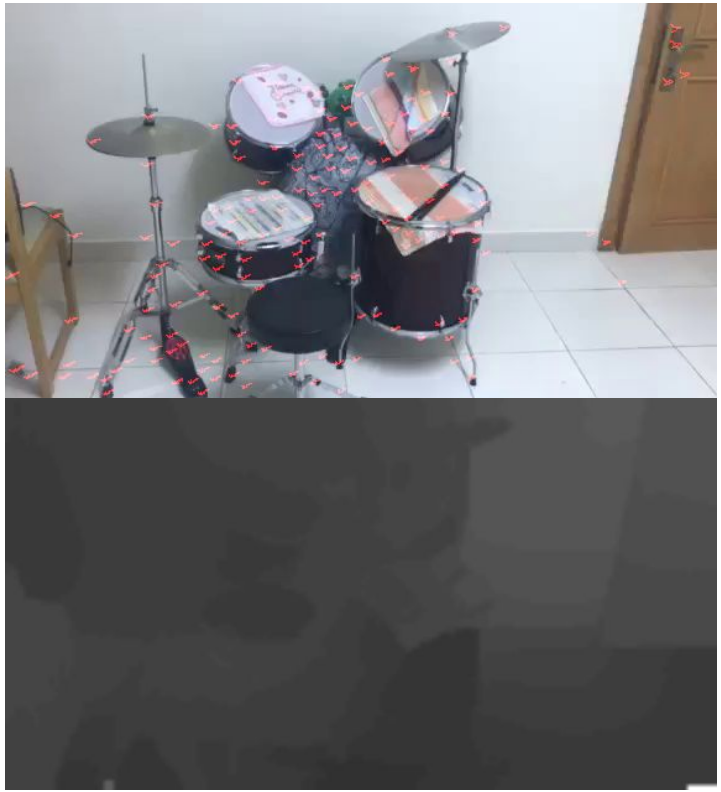
θ_c = 35

Results on Custom Images

Results



Results



Limitations

Limitations

- Since the bundle adjustment step uses the KLT tracker, there is a dependency on illumination and color of the images. It is observed that when there is spatial and illumination constancy, the results are better as compared to dimly lit scenes.
- The bundle adjustment for the pipeline requires the baseline between the images to be small in order to make crucial approximations to make the optimization well conditioned and convex. Therefore, the results depend on motion as well where in large motion does not ensure the inverse depth initialization to be convex.
- The region of interest in the object should be within a certain depth range in order to make the optimization convex and well conditioned.
- The plane sweeping unary potentials depend on the correct estimation of the extrinsic matrix. Therefore the number of images matter as they improve the bundle adjustment results. It is observed that as the number of images increases the KLT tracker extracts more confident feature points and the bundle adjustment improves the extrinsic matrix.

End Of Presentation

Energy Minimization

The energy , $E(D) = E_p(D) + \alpha E_s(D)$, is minimized using a dense CRF model which uses mean field approximation to give a smooth depth map.

- Mean field approximation uses an iterative message passing algorithm for approximate inference.
- The message passing in the CRF model is done using Gaussian filtering in some arbitrary feature space.
- The truncated linear function is implemented as two convolutions of 1D box filtering.
- This allows the running time to be linear to the number of depth labels thus allowing fast inference.
- The output of the energy minimization step is the final dense depth map as required.