A Comparison Based Study On Depth Estimation of Monocular Endoscopic Images using Self-supervised Learning Methods

Ayantika Das
EE19D422
ee19d422@smail.iitm.ac.in

Nicky Nirlipta Sahoo
EE19S042
ee19s042@smail.iitm.ac.in

INDIAN INSTITUTE OF TECHNOLOGY,MADRAS

Abstract

High-quality 3D reconstruction from endoscopic videos, play an important role in many clinical applications, including surgical navigation where they enable direct registration of intra-operative video with pre-operative CT data. Depth estimation plays a crucial pre-requisite for 3D-reconstruction. In this work we present an experimental based study on self-supervised training scheme and novel loss design for depth estimation. We only need endoscopy video images and a multi view 3D reconstruction technique such as Structure from motion (SFM) technique to supervise learning in sparse manner. We have compared the proposed technique with natural single image based depth estimation technique and have shown the performance improvement of the proposed method over it.

Index Terms -Endoscopy, SFM, sef-supervised, depth estimation

1. Introduction

Back Ground

Endoscopy is a Minimally Invasive Surgery (MIS) for examining and operating on the medical condition. It benefits patients through smaller trauma, shorter hospitalization, lesser pain and lower risk of infection than traditional open cavity surgeries. A device called endoscope is inserted into the body through a natural orifice. MIS provides surgeons with additional anatomical and positional information. This helps them avoid critical structures, such as the brain, eyes, and major arteries, that are spatially close to the sinus cavities and must not be disturbed during surgery. Computer vision-based navigation systems that rely on the intra-operative endoscopic video stream and do not introduce additional hardware are both easy to integrate into clinical workflow and cost-effective, but require registration of pre-operative data, such as CT scans, to the intra-operative videos [5]. For 3D-to-3D registration algorithms, estimating an accurate and dense intra-operative 3D reconstruction is necessary to ensure acceptable performance of the system. However, obtaining such reconstructions is not trivial due to problems such as textureless surface, specular reflectance,lack of photometric constancy across frames, and tissue deformation.

For 3D reconstruction, Shape from Shading based methods [7] model the relationship between appearance and depth but usually oversimplify the problem, which can result in inaccurate reconstructions in cases with, e.g., specular reflection. Hardware-based solutions such as depth or stereo cameras , are still challenging because of non-Lambertian properties of tissues and the paucity of features. Deep-learning based methods have recently been explored to solve the single-frame dense 3D reconstruction task in monocular endoscopy. Simulation-based works use synthetic dense depth maps generated from patient specific CT [8] to solve the problem of unpaired data. Additionally, the moving light source, lack of texture in the visible tissues, large specularities, and high dynamic range all contribute to difficult feature matching.Finally, the principal axis of motion inside the sinus cavity is along the optical axis of the endoscope, meaning that feature matching must be particularly robust across scales. Mahmood *et al*. [8], present a method that avoids these restrictions, using a joint deep convolutional neural network-conditional random field (CNN-CRF) framework. Estimated depth is used to reconstruct the topography of the surface of the colon from a single image. It trained the unary and pairwise potential functions of a CRF in a CNN on synthetic data, generated by developing an endoscope camera model and rendering over 100,000 images of an anatomically-realistic colon and validate this approach with real endoscopy images from a porcine colon, transferred to a synthetic-like domain, with ground truth from registered computed tomography measurements.The limitations of this method are artifacts due to specular reflections, cases where inverse of intensity might not be the

major cue and instances where the pairwise similarities can give rise to artifacts.A self-supervised method has been proposed by Liu *et al*. [6] that only requires unlabeled endoscopic videos.

Related Work and Contributions

To address the above challenges, a patient-specific learning-based method for 3D sinus surface reconstruction from endoscopic videos is being proposed by Liu *et al*. [6] is being demonstrated here. Multi-view stereo method based technique, Structure from Motion (SfM) [1] is used which is able to simultaneously reconstruct 3D structure while estimating camera poses in feature-rich scenes. The paucity feature in endoscopic images of anatomy can cause sparse production and unevenly distributed reconstructions, which is eradicated by removing these kind of 3D points using point cloud refinement technique. In endoscopy where no lighting consistency is available, capturing long-range correlation in either spatial or temporal domain is difficult which can be overcome by using SFM [1] which is capable of capturing long-range correspondences with illumination-invariant feature descriptors e.g., Scale-Invariant Feature Transform (SIFT), and global optimisation e.g., bundle adjustment. This method only requires monocular endoscopic images during both training and application phases. Several network loss function is proposed for integrating information from SFM and enforced geometric constraints between depth prediction of different view points. For depth estimation FC Dense Net model is being used as proposed by [4]. Cyclical learning rate is demonstrated as scheduler for the training model [9]. Finally a comparison based on the method proposed by [6] and natural image based endoscopy single image depth estimation as proposed in [2] is being done .
Method in terms of data preparation, network architecture, and loss design is described in Section II. Experimental setup , results with a comparison based study of proposed method as in [6] and natural image based single image depth estimation model [2] are demonstrated in Section III . In Section IV and V, the limitations of the proposed method and future directions to explore are described.

2. Methods

This section describes how SFM [1] is being implemented to extract self supervisory signal for estimating the depth. Depth estimation model along with loss function, Optimizer, scheduler, custom layers are is included which explains the training of the . The custom layers are being explained in network architecture section.

2.1. Feature Extraction

Features from video frame images is being extracted using Structure from Motion(SFM) [1] and pair wise Sparse depth map, Sparse flow map and Sparse Soft mask are being calculated between frames in adjacent range. The Block Diagram of Feature extraction is as shown in figure [2] .

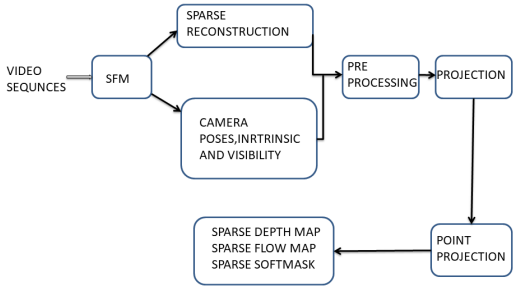

Figure 1: Block diagram of Feature Extraction From video sequences using SFM

SFM

SFM [1] is able to capture long range correspondences with illumination invariant feature descriptor (e.g.-SIFT) and global optimisation (e.g.- Bundle Adjustment). Using SFM 3D point cloud [3] and camera poses (e.g.-Positions And Orientations) are computed using sequential and global methods. In sequential method, first the structure problem is solved at local scale using minimum reconstruction based on 2-3 views using corresponding feature descriptor SIFT and RANSAC method .Then Bundle Adjustment is used for optimisation with sparse matrices and divide and conquer approach is used to reduce the computation. Finally the relative motions are being merged in to common global co-ordinate system.

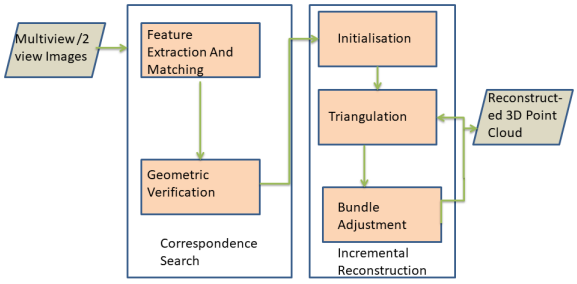

Figure 2: Block diagram of 3D Reconstruction using SFM

2.1.1 Preprocessing

In this step, Extrinsic and Projection Matrices are calculated using positional and orientational information from SFM [1] 3D point reconstruction for each image. From these , mask images, sparse depth images, flow mask images, flow images, are being calculated from a pair of images. Then relative motion between two frames are calculated, from which relative translations and rotations can be determined between pair of images. Also outliers are removed using Point Cloud Filtering.

2.1.2 Sparse Depth Map

For valid loss calculation, the scale of the depth prediction and the SfM results must match . So, Sparse depth map is used for scaling the depth prediction in the Depth Scaling Layer. In this step, 3D points estimated from sparse reconstruction using SFM, are projected on to image planes with camera poses, intrinsics and point visibility to generate sparse depth map. The camera pose of frame j with respect to the world coordinate(w) is T^j_w. The homogeneous coordinate of nth 3D point of the sparse reconstruction in the world coordinate is p^w_n, where n can be the index of any point in the sparse reconstruction. Frame indices used in the following equations, e. g., j and k, can be any indices within the same video sequence. The difference of j and k is kept within a specified range to keep enough region overlap. The coordinates of n^{th} 3D point w.r.t. frame j, p^j_n, is

$$p^j_n = T^j_w p^w_n \tag{1}$$

The depth of n^{th} 3D point w.r.t. frame j, z^j_n, is the z-axis component of p^j_n. The 2D projection location of n^{th} 3D point w.r.t. frame j, u^j_n is,

$$u^j_n = K\frac{p^j_n}{z^j_n} \tag{2}$$

where K is the intrinsic matrix. Using $b^j_n = 1$ to indicate that n^{th} 3D point is visible to frame j and $b^j_n = 0$ to indicate otherwise, the sparse depth map of frame j, z^s_j is,

$$z^s_j(u^j_n) = \begin{cases} z^j_n & if b^j_n = 1 \\ 0 & if b^j_n = 0 \end{cases} \tag{3}$$

2.1.3 Sparse Flow Map

Sparse Flow Map is used for estimating sparse flow loss. It combines camera trajectory, sparse reconstruction and all pair wise frame combinations to estimate 2D projected movement of sparse reconstruction from one frame to other and hence error distribution for one frame is more likely unbiased unlike sparse depth map which is fixed and potentially biased for one frame. The sparse flow map $F^s_{j,k}$, sparse reconstruction from frame j to frame k is defined as,

$$F^s_{j,k}(u^j_n) = \begin{cases} \frac{u^k_n - u^j_n}{(W,H)^T} & if b^j_n = 1 \\ 0 & if b^j_n = 0 \end{cases} \tag{4}$$

Where H and W are height and width of the frame respectively.

2.1.4 Sparse Soft Mask

Sparse soft mask is used to enable the network to exploit the valid sparse signals in the sparse from data and mitigate the effect of outliers in the 3D reconstructions. This can be achieved by assigning confidence values to valid regions in the image while masking out invalid regions. Valid regions are 2D locations on image planes where 3D points project onto, while the remaining image comprises invalid regions. The sparse soft mask, M_j, of frame j is defined as,

$$M_j(u^j_n) = \begin{cases} c_i & if b^j_n = 1 \\ 0 & if b^j_n = 0 \end{cases} \tag{5}$$

where c_i is a weight related to the number of frames used to reconstruct 3D point i and the accumulated parallax of the projected 2D locations of this point in these frames. Intuitively, c_i is proportional to the number of frames used for triangulation and the accumulated parallax. Greater magnitudes of c_i reflect greater confidence and hence greater is the accuracy. Here we have considered c_i as, $c_i = 1 - exp^{-\sum_i b^i_n/\sigma}$, where i iterates all frames in the video sequence where SFM is applied and σ is the hyper parameter based on average number of frames used to reconstruct each sparse point in SFM.

2.2. Training Process

This phase emphasizes upon training of the model for depth estimation, optimisation and scheduler for better learning rate and increasing computational efficiency. Loss function illustration for better accuracy is also being described.

2.2.1 Depth estimation Model (About FC DenseNet)

DenseNet is based on the idea that if each layer is directly connected to every other layer in a feed forward pattern then network will be more accurate and easier to train.

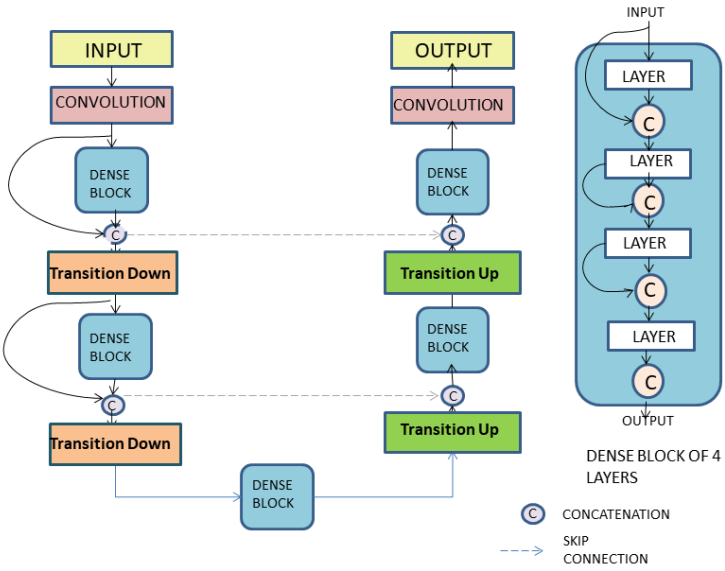

Figure 3: Dense net architecture with dense blocks description

Dense Net built from dense blocks and pooling layer has following advantages :

- Parameter Efficiency
- Deep Supervision
- Feature Reuse

The details of dense net architecture along with dense block description is as shown in the figure 3.

DenseNet design a connectivity pattern that iteratively concatenate all feature output in feed forward fashion. If x_l is the output of L^{th} layer ,then it is defined as

$$x_l = H_l([x_0, x_1, ...x_{l-1}]) \tag{6}$$

Where $[x_0, x_1, ...x_{l-1}]$ is a single tensor constructed by concatenation of previous layers. H_l is the non-linear transformation which is defined as a combination of Batch Normalisation (BN) followed by a rectifier non-linearity (ReLU), a convolution and drop out. Such connectivity pattern strongly encourages the reuse of features and makes all layers in the architecture receive direct supervision signal. The output dimension of each layer has k feature maps, where k is referred as growth rate parameter, is typically set to a small value (e.g. k = 12). Thus, the number of feature maps in DenseNets grows linearly with the depth (e.g. after layers,the input $[x_0, x_1, ...x_{l-1}]$ will have $l \times k$ feature maps). If input x_0 with m feature map generate output of dimension K by applying $H_l(x_0)$ then these k feature map concatenate to previous m feature map and act as an input to the second layer. Repeating this operation for n times will lead to a new dense block with $n \times k$ feature maps. Building Blocks of fully convolutional Dense Net are as shown in the figure 4.

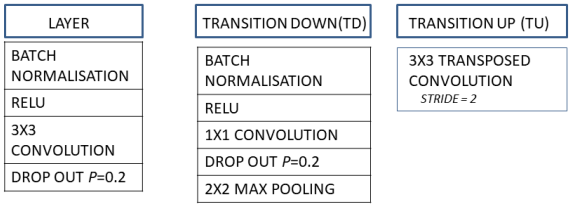

Figure 4: FC Dense net Bulding Blocks

Dense Net architecture consists of a down sampling path , an up sampling path.

- In the down sampling path, the linear growth in the number of features is compensated by the reduction in spatial resolution of each feature map after the pooling operation and the last layer of the down sampling path is referred to as *bottleneck*.
- Up sampling path is used to recover the input spatial resolution and it is composed of convolution ,up sampling operations(transposed convolution) and skip

connections. In FC DenseNet the convolution is substituted by *Dense block* and up sampling is referred to as *transition up* which is transposed convolution that up samples previous feature maps and to compensate the memory constraint it is only applied to last dense block. Skip connection helps up sampling path to recover fine grained information from down sampling path.

- Transition down is introduced to reduce the spatial dimensionality of the feature maps. Such transformation is composed of a 1×1 convolution (which conserves the number of feature maps) followed by a 2×2 pooling operation.

For given network, 57 layer dense net architecture is being used and the git code for FC Dense net is given in [1]

2.2.2 Optimization and Scheduler

For training the network Stochastic Gradient Descent (SGD) is used for optimisation and Cyclic learning rate is used as scheduler [9].

Cyclic Learning Rate - This method helps the learning rate vary cyclically between reasonable boundary values and also eliminates the need to tune the learning rate yet achieves near optimal classification accuracy. A cyclical learning rate is a policy of learning rate adjustment that increases the learning rate off a base value in a cyclical nature. Typically the frequency of the cycle is constant, but the amplitude is often scaled dynamically at either each cycle or each mini-batch iteration.

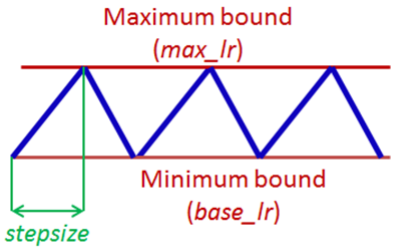

Figure 5: Triangular learning rate policy

The general structure of cyclicLR policy algorithm [9] is as given below,

```
local_cycle= math.floor(1+epoch_Counter/ (2*step_size))
local x = math.abs( epoch_Counter / step_size
- 2*cycle + 1)
local LR = opt.LR +(maxLR - opt.LR)*math.max(0,(1-
x))*scale_fn
```

where x is either iterations or cycle, depending on scale_fn mode. opt.LR is defined as lower learning rate and is denoted as base_lr. *epoch_Counter* is the number of epochs in training, lr is the computed learning rate, *step_size* is half the period or cycle length and *max_lr* is the maximum learning rate boundary. *scale_fn* may be defined as traingular, triangular 2 or exp_range. In traingular, the scale_fn is equal to 1. Trainguar 2 is is a triangular cycle that decreases the cycle amplitude by half after each period, while keeping the base lr constant and is given as $1/(2^{x-1})$. exp_range is a triangular cycle that scales the cycle amplitude by a factor $\gamma^{iterations}$, while keeping the base_lr constant . Code for this sheduler is given as [2]

2.2.3 Loss Functions

In the training phase two loss functions are being used to leverage the sparse depth annotation and relative camera poses between frames produced by SFM by enforcing geometric consistency between two frames.

1. Sparse Flow Loss(SFL) Network is trained to minimize the differences between dense flow maps and corresponding sparse flow maps to produce dense depth maps that agree with sparse reconstruction from SFM. It solves data imbalance problem caused arbitrary scales of SFM results by the 2D projected movements in the unit of pixels.This is scale invariant. SFL associated with frame j and k is,

$$L_{flow}(j,k) = \frac{1}{\sum M_j}\sum(M_j\left|F^s_{j,k} - F_{j,k}\right|) + \frac{1}{\sum M_k}\sum(M_k\left|F^s_{k,j} - F_{k,j}\right|) \quad (7)$$

   where $\sum_{M_j}$ is the summation of sparse masks, $F^s_{j,k}$ is the 2D projected movement of sparse reconstruction from frame j to frame k and $F_{j,k}$ is the 2D displacement field from frame j to frame k

2. Depth Consistency Loss(DCL) It adds spatial constraints among frames in the training phase. By using SFL only, the network does not gain any information where no sparse depth annotations are available and the training is prone to over fitting to the measurement noise or outliers from SFM -methods. The depth consistency loss helps gain more information and mitigate the over fitting issues. It requires inputs from Depth Map Scaling Layer and Depth Map Wrapping Layer.

[1] url for code of FC Dense Net `https://github.com/SimJeg/FC-DenseNet`.

[2] url for code of cyclical learning rate `https://github.com/bckenstler/CLR`.

$$L_{consist}(j,k) = \frac{\sum W_{j,k}(z_j - \tilde{z}^2_{k,j})}{\sum(W_{j,k}(z_j^2 + \tilde{z}^2_{k,j})} + \frac{\sum W_{k,j}(z_k - \tilde{z}^2_{j,k})}{\sum(W_{k,j}(z_k^2 + \tilde{z}^2_{j,k})} \quad (8)$$

where Z_j is the predicted depth map of frame j; $\tilde{z}_{k,j}$ is the warped depth map warped from its original frame j to frame k; $W_{k,j}$ is the intersection of valid regions of predicted depth map Z_j and dense depth map $\tilde{z}_{k,j}$.

Finally, the **Overall Loss** for network training of single pair of training data from frame j to frame k is the weighted sum of SFL and DCL defined as,

$$L(j,k) = \lambda_1 L_{flow}(j,k) + \lambda_2 L_{consist}(j,k) \quad (9)$$

2.3. Overall Network Architecture

The training network is a two-branch Siamese network. The input is a pair of color images, which are used as source and target. The training goal is, given a key point location in the source image, finding the correct corresponding key point location in the target image. A SFM method [1] with SIFT is applied to video sequences to estimate the sparse 3D reconstructions and camera poses. The sparse depth maps generated from SFM are used as ground truth. In the application phase, the network has a simple single-branch architecture for depth estimation from a single frame. All the custom layers described below are differentiable so that the network can be trained in an end to end manner.

2.3.1 Depth Scaling Layer

It scales the predicted depth map from the single frame depth estimation architecture to the corresponding SFM result to remain consistent with the scale of the coordinate transformation and correct loss calculation. The scaled depth prediction of frame j, Z_j is,

$$Z_j = (\frac{1}{\sum M_j} \sum M_j \frac{Z_j^s}{Z_j' + \epsilon}) z_j' \quad (10)$$

where z_j' is the depth prediction of frame j that is correct upto a map; $\sum M_j$ is the summation of sparse masks and ϵ is a hyper parameter to avoid zero division

2.3.2 Flow From Depth Layer

To guide network training with SFL, scaled depth map is converted to dense flow map with relative camera poses and intrinsic matrix. Dense flow map is used for depth estimation training. The dense flow map is basically the 2D displacement field describing 3D viewpoint change. Given the scaled depth map of frame j, the relative camera pose of frame k w.r.t frame j is, $T_j^k = (R_j^k, t_j^k)$. The 2D locations in frame j,(U, V), are organised as a regular 2D mesh grid. The corresponding 2D locations of frame k are (U_k, V_k), which are organized as same spatial arrangement as frame j.U_k, V_k is given by,

$$U_k = \frac{Z_j(A_{0,0}U + A_{0,1}V + A_{0,2}) + B_{0,0}}{Z_j(A_{2,0}U + A_{2,1}V + A_{2,2}) + B_{2,0}}$$
$$V_k = \frac{Z_j(A_{1,0}U + A_{1,1}V + A_{1,2}) + B_{0,0}}{Z_j(A_{2,0}U + A_{2,1}V + A_{2,2}) + B_{2,0}} \quad (11)$$

U consists of H rows of $[0, 1, ...W-1]$ and V consists of W columns of $[0, 1, ...H-1]$. $A = KR_j^k K^{-1}$ and $B = -Kt_j^k$. $A_{m,n}$ and $B_{m,n}$ are elements of position(m, n), respectively.The dense flow map , $F_{j,k}$,for describing the 2D displacement field from frame j to frame k is,

$$F_{j,k} = (\frac{U_k - U}{W}, \frac{V_k - V}{H}) \quad (12)$$

2.3.3 Depth Wrapping Layer

The Depth Map Warping Layer warps a scaled dense depth map to the coordinate frame of the other input to the Siamese network using the relative camera pose between the two frames. The guidance to regions of frames where sparse information from SFM projected, provided by sparse flow map, only have a small percentage of pixels whose values are valid in sparse flow map. So most of the regions are not properly guided. By enforcing consistency between two corresponding depth predictions with the camera motion and camera intrinsics, geometric constraint between two frames can be exploited. To make the geometric constraint in DCL differentiable, the view points of of depth predictions have to be aligned. As dense flow map describes 2D projected movement of 3D scene, So U_k and V_k can be used to change view point of frame k to frame j by modifying depth map of frame k ,$\tilde{z}_k$ is ,

$$\tilde{z}_k = z_k(C_{2,0}U + C_{2,1}V + C_{2,2}) + D_{2,0}, \quad (13)$$

where, $C = KR_k^j K^{-1}$ and $D = Kt_k^j$. Bilinear sampler can generate Dense depth map $\tilde{z}_{k,j}$ wrapped from the view point of frame k to frame j with U_k, V_k and $\tilde{z}_k$.

3. Experiment And Results

3.1. Experimental Requisites

All the experiments were carried out in Google Colaboratory with NVIDIA Tesla K80 GPU with 12 GB mem-

ory and 64 GB Disk space. The methods were carried out in Python 3 with PyTorch implementation framework.The data set used for experimentation has 400 image frames from monocular rectified sinus endoscopy videos. The entire data set is splitted up for training, validation and testing in the ratio of 7:1:2. Since, our method lacks ground truth, data generation , valid loss calculation plays a vital role. So, parameters responsible for training-data generation in addition to training parameters for the depth estimation model has to be tuned for optimal outcome. Our training parameters were tuned based on validation outcomes and training-data generation parameters were dependent on the endoscopic video chosen.

3.2. Optimal Parameters

- Training-data Generation: The number of projected points per image, are filtered using point visibility information from SfM reconstruction to remove the outliers. Several other factors constrain the number of visible points per frame, like visibility continuity, minimum and maximum sanity threshold, height and width of the image frames. By exploiting the continuity of camera movement within frames, visibility is smoothed up-to a factor of 30. This smoothing process removes occluded points that lies within the visibility range. The minimum and maximum sanity threshold is evaluated by calculating the frequency of occurrence of the brightness values and the depths of the projected points.Lower frequency values are clipped off, considering them as outliers. The average minimum and maximum sanity threshold values are 0.25 to 0.33. Since, all images extracted from the videos are cropped to remove the invalid blank regions and down sampled to the resolution of 256X320. So, points out of this range are considered as outliers.The sparsity accounted for in each image is approximately about 2%.The Point cloud overlay generation of the endoscopy data is as shown in figure 6.

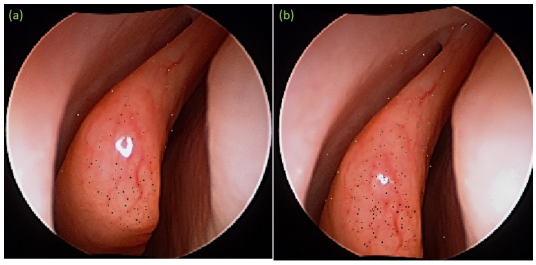

Figure 6: Filtered sparse reconstruted points ovelaid on video frames

- Training Parameters: The frame interval of two frames that are randomly selected from the same sequence and fed to the two-branch training network is set to [5, 30]. We use extensive data augmentation during experiments to make the training data distribution unbiased to specific patients or cameras as much as possible, e.g., random brightness, random contrast, random gamma, random HSV shift, Gaussian blur, motion blur, jpeg compression, and Gaussian noise. During network training, we use Stochastic Gradient Descent (SGD) optimization with momentum set to 0.9 and cyclical learning rate scheduler [9] with learning rate from $1.0e^{-4}$ to $1.0e^{-3}$. The batch size is set to 5. The for generating the soft sparse masks is set to the average track length of points in the sparse reconstructions from SfM. The ϵ in the depth scaling layer is set to $1.0e^{-8}$.We train the network with 80 epochs in total.1 is always 20. For the first 20 epochs, 2 is set to 0.1 to mainly use SFL for initial convergence. For the remaining 50 epochs, 2 is set to 5.0 to add more geometric constraints to fine-tune the network.

3.3. Results

Figure 7 depicts the training and validation curves. We have chosen optimal overall validation loss as 0.096, which corresponds to an optimal overall training loss of 0.08. The curves are of expected nature.They have a decreasing pattern, with highly varying slopes in the initial phase and then reaches saturation. It is ensured that, the optimal parameters are chosen in the saturation regions only. Here, the sparse flow loss significantly controls the overall loss and the depth consistency loss is less dominant in nature. Figure 8 represents colour images, predicted depths, sparse flow maps & dense flow maps.

Figure 7: Loss curves for Train and Vailadation

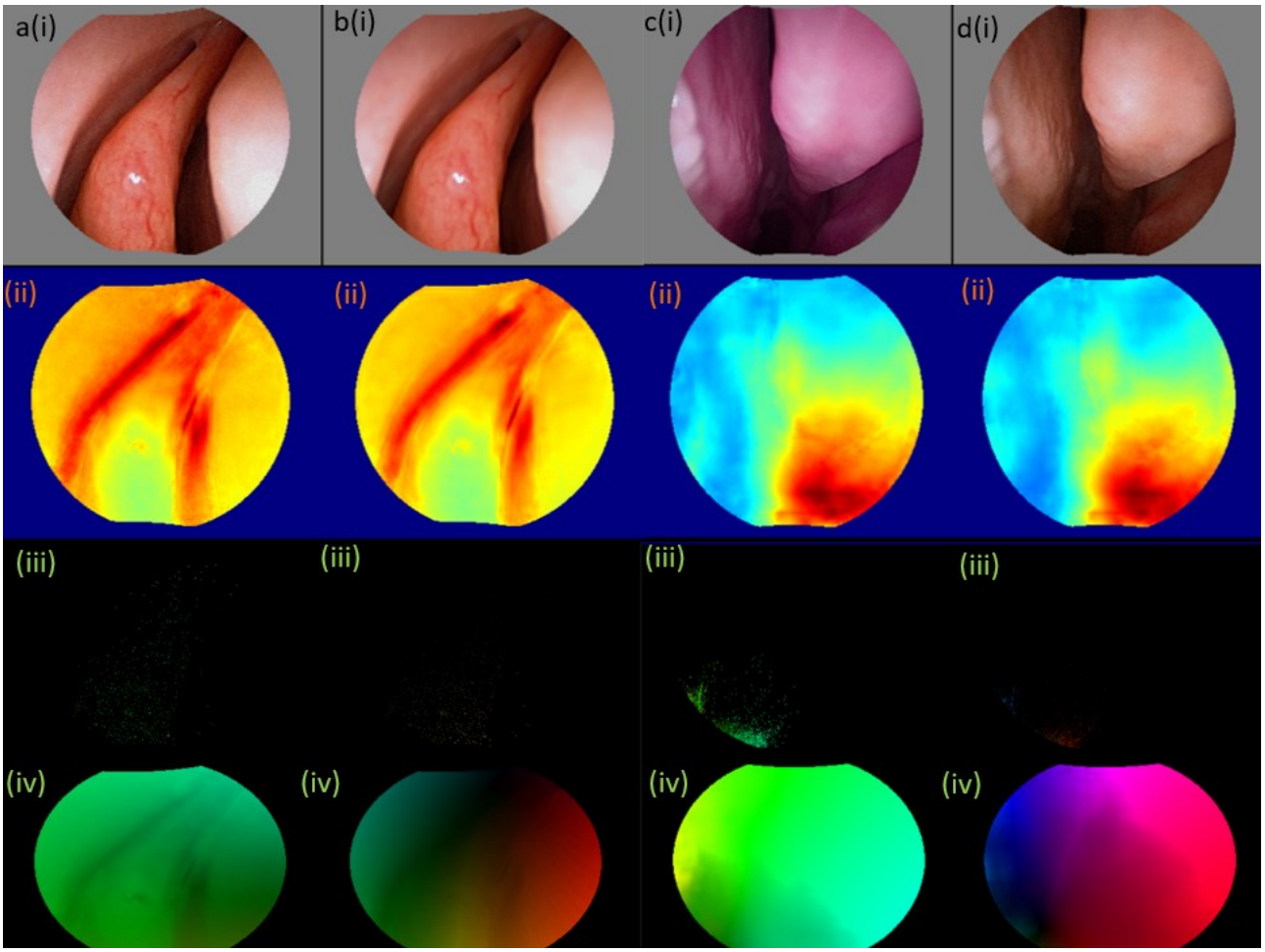

Figure 8: Results of our experiment -Images a-b(i) & c-d(i) are adjacent frames from train set and test set respectively.The first row consists of testing and training images, where a & c training and b & d testing ones. The second row consist of corresponding depth maps. The third row shows the sparse flow maps and the fourth row is the corresponding dense flow maps.

4. Comparative Study

4.1. Comparison with trained model

We have conducted a comparative study to evaluate the performance of our implemented method against one of the most recent methods of monocular self supervised depth estimation methods [2]. Clément *et al*. [2] uses a monocular depth estimation approach that robustly handle occlusions. It is a full-resolution multi-scale sampling method that reduces visual artifacts, and also incorporates an auto-masking loss to ignore training pixels that violate camera motion assumptions. The model is pre-trained on benchmarking depth estimation data set, KITTI data set. Figure 9 represents qualitative results for both the methods. It is a transparent portrayal, that our implemented method has outperformed in both test & training case. The study in both the cases, proves that better performance of our method can't be attributed to over-fitting of our model.

4.2. Comparison with two training data

We have evaluated our method with two different rectified sinus endoscopy videos (V1 & V2). Figure 10 depicts the depth estimation for two adjacent frames from each of the videos. Clearly, frames (a) & (b) from video V1 outperforms frames (c) & (d) from video V2. This performance variation is attributed to better SfM reconstruction for V1. Figure 10 a-b(iii) & c-d(iii) represents the sparse reconstruction in adjacent frames of both the videos. Dense, cluttered and unevenly distributed sparse representation for c-d(iii) along with poor representability is a clear indication

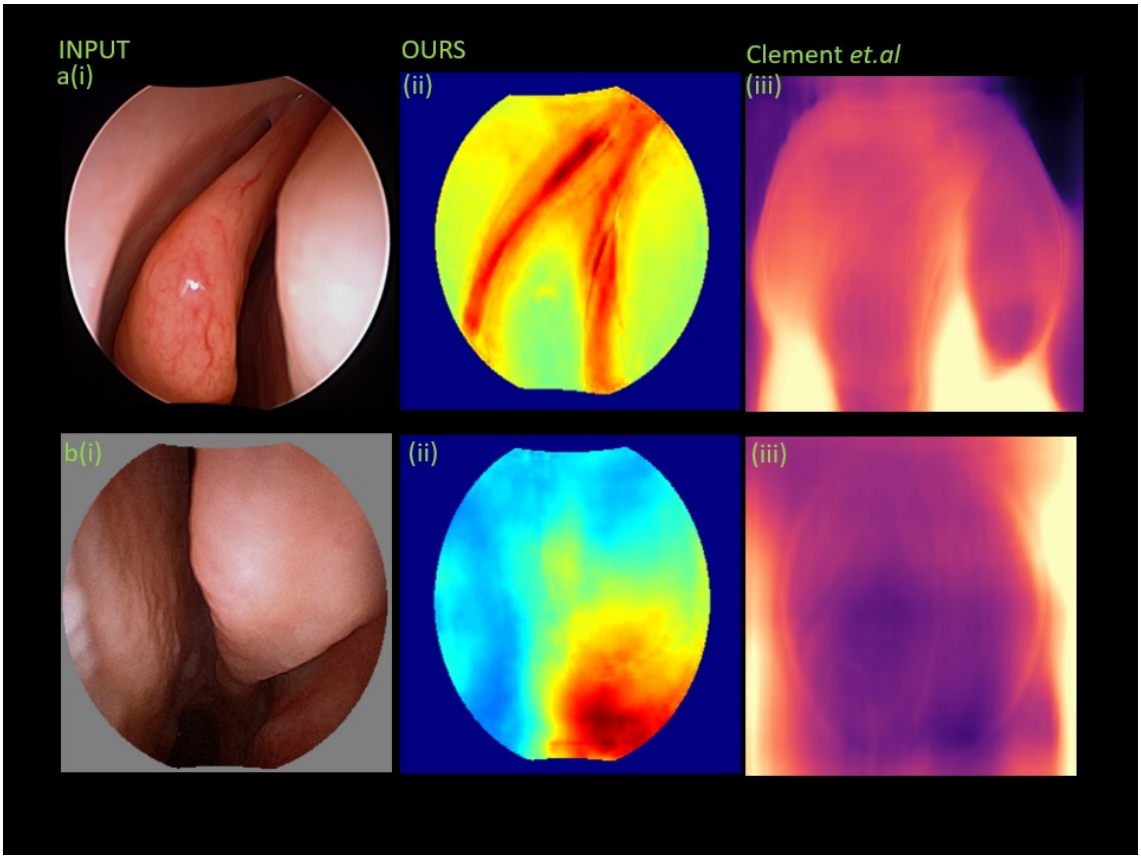

Figure 9: Qualitative result comparison between our method and Clément *et al*. [2]. The first column shows the images from data set 1 and 2, The second column shows the predicted depth of the corresponding images of our method and the third column shows the predicted depth from Clément *et al*.

that inappropriate sparse flow map has been generated from SfM point cloud. Although SfM used for 3D reconstruction is tolerable to errors upto a certain extent, but sparse flow map clearly indicates quite large systematic errors in SfM point cloud generation. These errors were subject to highly dynamic capturing environments.

5. Discussion

The proposed method is independent of any data labelling. We only require monocular endoscopic videos for testing. Our results quite evidently show that the trained model could very efficiently learn the depth estimation task from the sparse points generated by SfM, along with position, orientation and filtering information collected from SfM. It is also quite evident that common challenges in endoscopic videos like lack of photometric constancy & texture information, alongside repetitive and smooth contents are well addressed by our depth estimation model. The model being independent of registration task elevates registration dependent errors completely. Although the algorithm does not explicitly use any loss for enforcing the geometric consistency applicable for depth estimation of two adjacent frames but uses the Depth consistency Layer which is able to handle errors in the depth predictions. In case of the first comparison, the pre-trained model on KITTI dataset has failed due to variation in outdoor scenario (used by a self driving car) as compared to endoscopic scenes. Several factors like illumination variation in endoscopic scenes, the significant change in appearance of the anatomical structures within adjacent frames, texture-scarce, repetitive and smooth contents makes it pretty challenging task for such pre-trained model to perform depth estimation on endoscopic videos. The second comparison well portrays the significance of SfM reconstruction for our model's depth prediction.The comparison results show-case how the depth predictions fail in cases where, collection of valid SfM reconstructed points are insignificant. In such cases, SfM points fail to be a representation of the underlying structure within the captured video and thus we failed

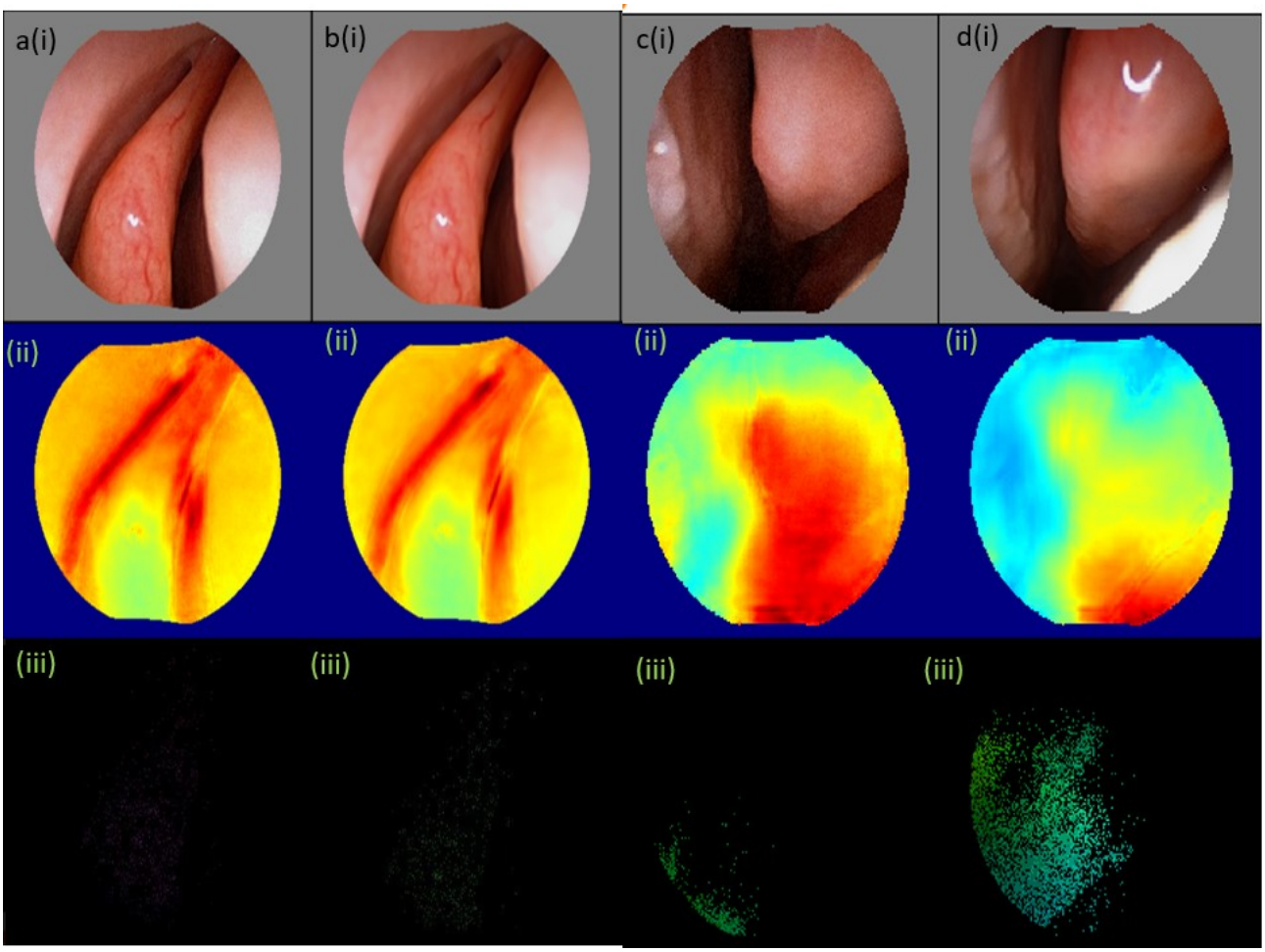

Figure 10: comparison between Video V1 and V2.The first row consists of testing and training images from V1 & V2, where b & d are testing ones. The second and third columns consist of corresponding depth maps and sparse flow maps.

to train our model in such scenarios.

6. Conclusion

In this experimental study we have implemented a self-supervised approach to train a convolutional neural network for depth estimation of monocular endoscopy videos. The implemented method is the first of its kind, deep learning-based self-supervised depth estimation method proposed for monocular endoscopy videos. We could emulate the results claimed by the proposed method in their original implementation. Our results also stand in favour of the drawbacks the method is claimed to have. In one comparison study, we have used a dataset which has constraints on training-data generation and were able to outcast poor results as anticipated. The other comparison study, has shown that our implemented model outperforms one of the most competitive method. For, future work we can implement the method with algorithms that can carry out exhaustive search for correspondences in critical datasets, such that SfM reconstruction performs better. This can aid to more generalising capability of the method.

7. Acknowledgement

The authors would thank Dr. Mansi Sharma (Department of Electrical Engineering, IIT Madras) for guiding through out the project and her insightful feedback on the progress of the project. Thanks to the authors who have shared their results.

References

[1] *SFM Reconstruction*, 2008. `http://mi.eng.cam.ac.uk/~cipolla/`

publications/contributionToEditedBook/2008-SFM-chapters.pdf.

[2] Clement Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. October 2019.

[3] Qi Hu, Jianxin Luo, Guyu Hu, Weiwei Duan, and Hui Zhou. 3d point cloud generation using incremental structure-from-motion. 1087(6), 2018.

[4] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. pages 11–19, 2017.

[5] Simon Leonard, Ayushi Sinha, Austin Reiter, Masaru Ishii, Gary L Gallia, Russell H Taylor, and Gregory D Hager. Evaluation and stability analysis of video-based navigation system for functional endoscopic sinus surgery onin vivoclinical data. *IEEE transactions on medical imaging*, 37(10):2185–2195, 2018.

[6] X. Liu, A. Sinha, M. Ishii, G. D. Hager, A. Reiter, R. H. Taylor, and M. Unberath. Dense depth estimation in monocular endoscopy with self-supervised learning methods. *IEEE Transactions on Medical Imaging*, 39(5):1438–1447, 2020.

[7] Xingtong Liu, Ayushi Sinha, Mathias Unberath, Masaru Ishii, Gregory D Hager, Russell H Taylor, and Austin Reiter. Self-supervised learning for dense depth estimation in monocular endoscopy. pages 128–138, 2018.

[8] Faisal Mahmood and Nicholas J Durr. Deep learning and conditional random fields-based depth estimation and topographical reconstruction from conventional endoscopy. *Medical image analysis*, 48:230–243, 2018.

[9] Leslie N Smith. Cyclical learning rates for training neural networks. pages 464–472, 2017.