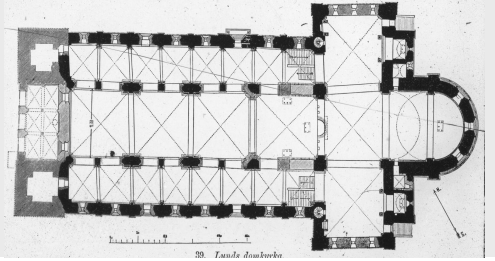
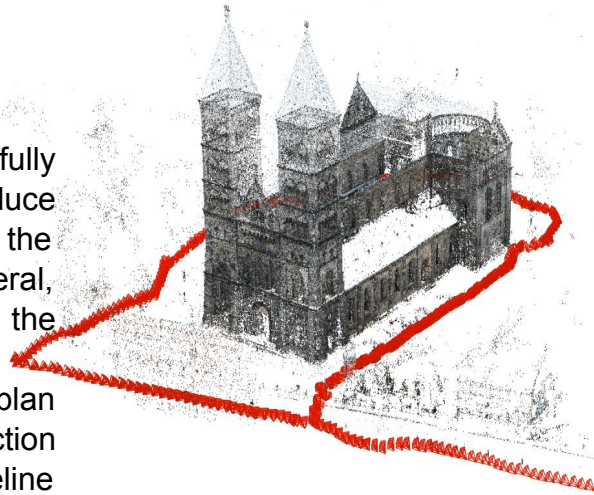
2D-Map Guided Incremental SfM



Goal: Leverage 2D-maps (e.g. floor plans) in an incremental Structure-from-Motion pipeline



Description:

Incremental Structure-from-Motion (SfM) seeds the reconstruction starting from two carefully selected views, and iteratively registers new images and triangulates new points to produce a sparse reconstruction. Finally, a global Bundle Adjustment improves the quality of the reconstruction by minimizing the overall reprojection error [1]. Although effective in general, this process does not integrate a-priori information available on the subject of the reconstruction.
Nowadays most buildings come with an accurate metric floor plan available. This floor plan and top-view 2D maps in general can strongly constrain the image-based reconstruction process, addressing limitations such as scale ambiguity, global drift, and short baseline triangulation uncertainty. The task of this project is to integrate 2D top-view map prior information within the Incremental SfM pipeline. This can be achieved as an integration of Colmap, a state-of-the-art open source incremental SfM software [2].

[1] Schonberger, Johannes L., and Jan-Michael Frahm. "Structure-from-motion revisited." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[2] https://github.com/colmap/colmap

Requirements / Tools:

C++

Supervisor:

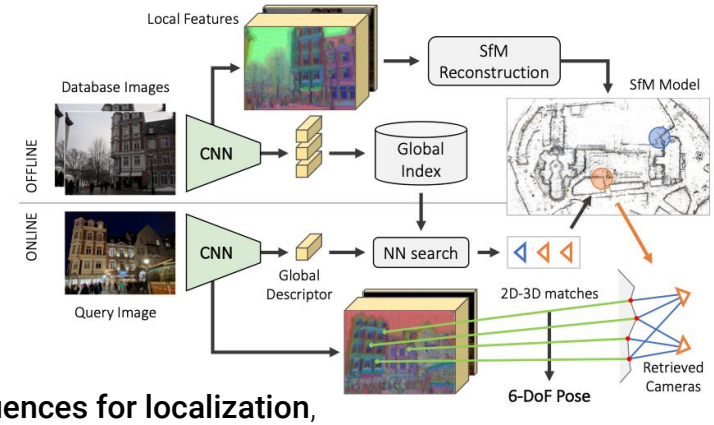
Mihai Dusmanu <mihai.dusmanu@inf.ethz.ch>
Luca Cavalli <luca.cavalli@inf.ethz.ch>
Viktor Larsson <vlarsson@inf.ethz.ch>

Sequence & multi-camera visual localization

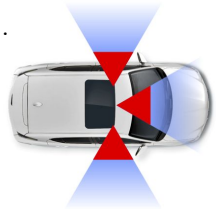
Goal: Develop new algorithms to localize sequences of images captured with multi-camera rigs

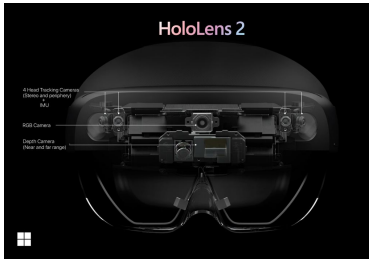
Description:

Successful visual localization approaches involve a combination of **image retrieval and local feature matching** [1]. They have all so far considered only **individual query images**. Systems deployed for practical applications, like **AR headsets or autonomous robots**, however record videos and often consist of rigs of multiple cameras facing different directions.



In this project, we want to develop new techniques to exploit **multi-camera sequences for localization**, in terms of both image retrieval and matching. This could involve retrieving rigs or sequences of global descriptors, and leveraging matches between images within a sequence, with bundle adjustment and generalized pose solvers. The final result will be **integrated into our popular library hloc** [2] and will try to push the state of the art on various benchmarks [3].





[1] Sarlin et al. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. CVPR 2019.
[2] hloc: https://github.com/cvg/Hierarchical-Localization
[3] Sattler et al. Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. CVPR 2018.

Requirements / Tools:

Proficiency in Python, willing to write some C++ if needed
Tools: hloc, Ceres solver

Supervisor:

Paul-Edouard Sarlin / psarlin.com / psarlin@ethz.ch
Viktor Larsson / vlarsson@inf.ethz.ch

Line-based SfM with relaxed endpoint constraints

Goal: Implement a Structure-from-Motion (SfM) pipeline with line features only.

Description:

Structure-from-Motion (SfM) is a well-studied problem with feature points, and can yield very accurate 3D reconstructions of real scenes from a set of raw images. However, points suffer from inherent limitations such as poor matchability in low texture area and they are sensitive to repeated structures. Reconstructing a scene based on lines instead of points can circumvent these drawbacks and additionally offer stronger geometrical constraints than single points.

However, the endpoints of line segments are often poorly localized, making it hard to base an accurate SfM system on line segments only. Recently, Branislav Micusik and Horst Wildenauer [1] proposed an SfM pipeline using line features only, by relaxing the constraints on the endpoints of the segments. This project aims at re-implementing the pipeline described in this paper to be able to perform line SfM.

[1] Branislav Micusik, Horst Wildenauer, *Structure from Motion with Line Segments Under Relaxed Endpoint Constraints,* IJCV 2017 (https://link.springer.com/article/10.1007/s11263-016-0971-9)

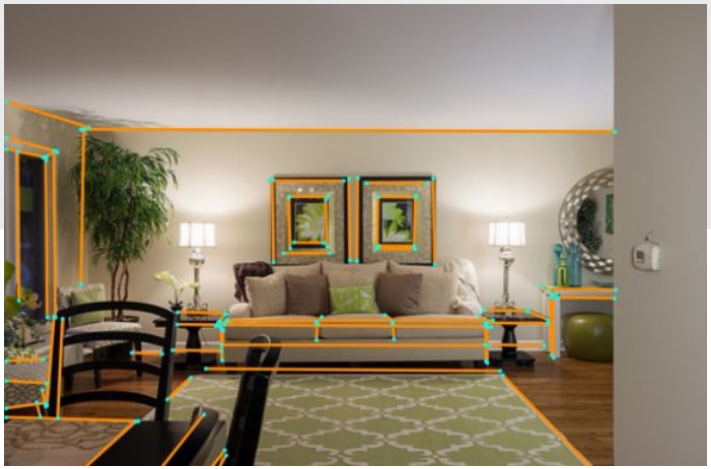
Requirements / Tools:

C++

Supervisor:

Rémi Pautrat <remi.pautrat@inf.ethz.ch >
Viktor Larsson <viktor.larsson@inf.ethz.ch >

Line Segment Detection with Transformers



Goal: Implement a deep line segment detector in images using transformers.

Description:

Line segment detection is a base step for numerous 3D applications such as 3D reconstruction, room layout estimation, accurate camera pose estimation, etc. Recent methods proposed to learn line detection with deep networks and displayed major improvements over handcrafted methods in terms of line repeatability. However the localization accuracy of these methods is still lagging behind handcrafted methods and most methods are limited to full supervision from hand-labelled lines.

In this project, we aim at leveraging the recently introduced transformers [1] to perform line segment detection, using the method described in [2]. The resulting model will be benchmarked against previous methods in terms of repeatability and localization error, in order to know the level of accuracy that transformers can achieve. Additional improvements may include exploring new ideas to depart from the fully supervised approach, and improving the initial network architecture.

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, *Attention Is All You Need*, NeurIPS 2017.
[2] Y. Xu, W. Xu, D. Cheung, Z. Tu, Line Segment Detection Using Transformers without Edges, ArXiv 2021.

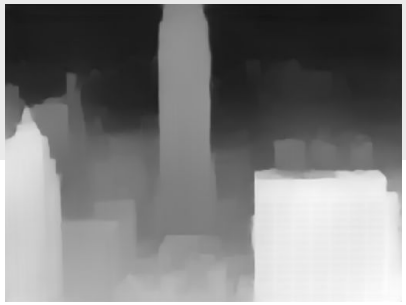
Requirements / Tools:

Python, knowledge in deep learning, Pytorch/Tensorflow

Supervisor:

Rémi Pautrat <remi.pautrat@inf.ethz.ch >

Plane-based Global SfM



Goal: Implement a global Structure-from-Motion (SfM) pipeline with plane-to-plane correspondences.

Description:

Global Structure-from-Motion (SfM) is a well-studied problem with feature points, and can yield very accurate 3D reconstructions of real scenes from a set of raw images significantly faster than incremental SfM techniques. However, the processing time of the feature matching and robust pose-graph initialization by RANSAC depends quadratically on the image and feature number. Thus, the initialization for the global SfM dominates the processing time.

The task is to avoid feature point matching by applying a monocular depth prediction network, finding multiple planes using the predicted relative depths and trying to match (i.e., estimate the relative pose) the images by creating plane-to-plane correspondences. Do the feature point matching only if no plane-to-plane correspondences are found. The implemented global SfM should triangulate the 3D points by projecting them to the assigned plane if such plane has been found.

[1] Li, Zhengqi, and Noah Snavely. "Megadepth: Learning single-view depth prediction from internet photos." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.
[2] Barath, Daniel, and Jiri Matas. "Progressive-X: Efficient, anytime, multi-model fitting algorithm." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
[3] Wilson, K. and Snavely, N. Robust Global Translation with 1DSfM European Conference on Computer Vision, 2014.

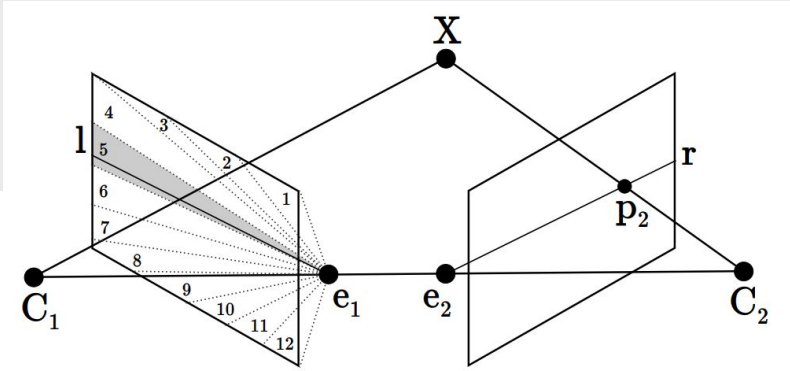
Requirements / Tools:

C++

Supervisor:

Daniel Barath <danielbela.barath@inf.ethz.ch >

Epipolar Hashing on Arbitrary Number of Images for Global SfM



Goal: Implement the Epipolar Hashing Algorithm for Guided Feature Matching on Arbitrary Number of Images

Description:

Global Structure-from-Motion (SfM) is a well-studied problem with feature points, and can yield very accurate 3D reconstructions of real scenes from a set of raw images significantly faster than incremental SfM techniques. However, the processing time of the feature matching and robust pose-graph initialization by RANSAC depends quadratically on the image and feature number. Thus, the initialization for the global SfM dominates the processing time.

The task is to implement a global SfM pipeline where the feature matching in the pose-graph initialization is sped up via extending the Epipolar Hashing algorithm to be applicable in the case when *more than two images* are to be matched. When having more than two images, the matching can be done as hashing where each bin is a polygon in the image. This is particularly useful for camera rigs, where the rig is calibrated a priori. In this case, when matching a new image to one of the images from the rig, the pose can be straightforwardly calculated between all view pairs without additional feature points. This is the case also, when a new image is to be matched to a connected component in the pose-graph.

[1] Barath, Daniel, et al. *Efficient Initial Pose-graph Generation for Global SfM.* arXiv preprint arXiv:2011.11986 (2020).
[2] Wilson, K. and Snavely, N. *Robust Global Translation with 1DSfM*, European Conference on Computer Vision, 2014.

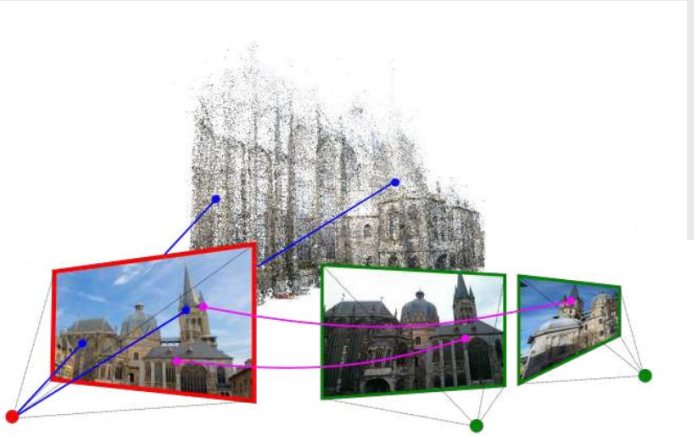
Requirements / Tools:

C++

Supervisor:

Daniel Barath <danielbela.barath@inf.ethz.ch >

Hybrid RANSAC++



Goal: Implement the Hybrid RANSAC algorithm with the state-of-the-art features of robust estimation

Description:

The task is to implement the hybrid RANSAC algorithm which uses multiple pose solvers applied to 2D-2D and 2D-3D correspondences to find the pose of an image given an existing reconstruction generated by, e.g., a Structure-From-Motion algorithm. The implementation should include the state-of-the-art features of robust estimation, e.g.,

- The scoring technique from MAGSAC++.
- Progressive NAPSAC sampling where the correspondences for the inner PROSAC sampler are ordered according to the scores predicted by NG-RANSAC.
- SPRT, degeneracy and cheirality tests.
- Local optimization by Graph-Cut RANSAC.

[1] Federico Camposeco, Andrea Cohen, Marc Pollefeys, Torsten Sattler; *Hybrid camera pose estimation*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 136-144
[2] Daniel Barath, Jiří Matas; *Graph-Cut RANSAC*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018
[3] Daniel Barath, Jana Noskova, Maksym Ivashechkin, Jiri Matas; *MAGSAC++, a fast, reliable and accurate robust estimator*, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 1304-1312
[4] Brachmann, Eric, and Carsten Rother. *Neural-guided RANSAC: Learning where to sample model hypotheses.* Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

Requirements / Tools:

C++

Supervisor:

Daniel Barath <danielbela.barath@inf.ethz.ch >

Soccer Ball Detection and Tracking (FIFA)

Goal: Apply and improve methods to detect and track a soccer ball from one or multiple overview cameras.

Description:

Apply methods for Fast Moving Objects detection [1,2] to estimate the ball 2D trajectory for the overview cameras. The main challenge is to deal with high amount of motion blur since the ball appears as a highly blurred streak.
The data is provided by FIFA as videos, which are stored in an interlaced mode (only every second image row is saved), and some interpolation techniques must be used.
Training data for learning based approaches [1] will be synthetically created, e.g. by using Google Research Football toolkit [3].
In the first stage, the ball trajectory will be estimated only in 2D for each camera independently. Then, the final goal is to get full 3D trajectory of the ball using all cameras. The methods for camera calibration will be given.

[1] Denys Rozumnyi, Jiri Matas, Filip Sroubek, Marc Pollefeys, Martin R. Oswald. “FMODetect: Robust Detection and Trajectory Estimation of Fast Moving Objects”, arxiv 2020
[2] https://github.com/rozumden/fmo-cpp-demo
[3] https://github.com/google-research/football

Requirements / Tools:

Python, PyTorch, Blender

Supervisor:

Denys Rozumnyi <denys.rozumnyi@inf.ethz.ch>
Martin Oswald <martin.oswald@inf.ethz.ch>

Time synchronization of multiple soccer streams

Goal: Develop a method for autonomous synchronization of multiple soccer streams.

Description:

The camera streams of soccer games from FIFA are not perfectly time synchronized. The goal is to automatically synchronize them, possibly with sub-frame accuracy. To achieve this, landmark events should be detected in each stream, such as ball bounces [1] or player poses [2], or even both. Then, a matching problem will be solved to temporally align all streams.
Synthetic data for training could be generated using simulators [3].

[1] Denys Rozumnyi, Jiri Matas, Filip Sroubek, Marc Pollefeys, Martin R. Oswald. “FMODetect: Robust Detection and Trajectory Estimation of Fast Moving Objects”, arxiv 2020
[2] Z. Cao et al. “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”, TPAMI 2019
[3] https://github.com/google-research/football

Requirements / Tools:

Python, PyTorch, Blender

Supervisor:

Martin Oswald <martin.oswald@inf.ethz.ch>
Denys Rozumnyi <denys.rozumnyi@inf.ethz.ch>

Sim2Real Video Style-Transfer for Soccer Simulation Videos (FIFA)

Goal: Create photo-realistic synthetic videos which will be used for algorithm testing (camera calibration, player tracking, ball tracking) with GT data.

Description:

The goal of this project is the creation of photo-realistic video stream of soccer matches that look like TV-broadcast video streams. Starting from rendered simulator output [1], the goal is to create and train a deep style-transfer approach to increase the realistic look of the videos. Particular steps will be as follows:

- Adapt the open-source soccer simulator [1] to add TV cameras with fixed positions (transferred from real data) while the camera orientation smoothly follows the ball.
- Create and train a deep network for unpaired video-to-video style transfer [2,3,4,5] between the simulator's output videos and real video sequences.
- Maybe add artificial motion-blur to improve realism.

Soccer video material with calibrated cameras will be provided.

[1] Google Research Football, https://arxiv.org/abs/1907.11180 , https://github.com/google-research/football
[2] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017
[3] J,-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, E. Shechtman, Toward Multimodal Image-to-Image Translation, NeurIPS 2017
[4] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, B. Catanzaro, Video-to-Video Synthesis, NeurIPS 2018
[5] Wei Gao, Yijun Li, Yihang Yin, Ming-Hsuan Yang, Fast Video Multi-Style Transfer, WACV 2020

Requirements / Tools:

Python, PyTorch, C/C++

Supervisor:

Martin Oswald <martin.oswald@inf.ethz.ch>

3D Player Position Tracking via Multi-view Streams (FIFA)

Goal: Develop a robust football player 3D position tracking method based on multiple-view soccer streams. (The data is provided by FIFA)

Description:

Intelligent sports video analysis systems have many commercial applications. Recently, with the emergence of accurate object detection and tracking algorithms, the focus is on a detailed analysis of sports videos, such as player tracking and identification [1]. However, accurate 3D player tracking like in football is challenging due to several reasons :
i: players are moving fast.
Ii: the players in the same team wearing the same shirts
Iii: overlappings and occlusions

The goal of this project is to leverage SOTA object detection, person tracking/ReID algorithms to deliver a robust player tracking framework in 3D based on multiple-view soccer streams.

[1] Lu, W. L., Ting, J. A., Little, J. J., & Murphy, K. P. (2013). Learning to track and identify players from broadcast sports videos. *IEEE transactions on pattern analysis and machine intelligence*, *35*(7), 1704-1716.

Requirements / Tools:

Python, PyTorch, Blender

Supervisor:

Martin Oswald <martin.oswald@inf.ethz.ch>
Jie Song <jsong@inf.ethz.ch>

3D Player Pose Estimation via Multi-view Streams (FIFA)

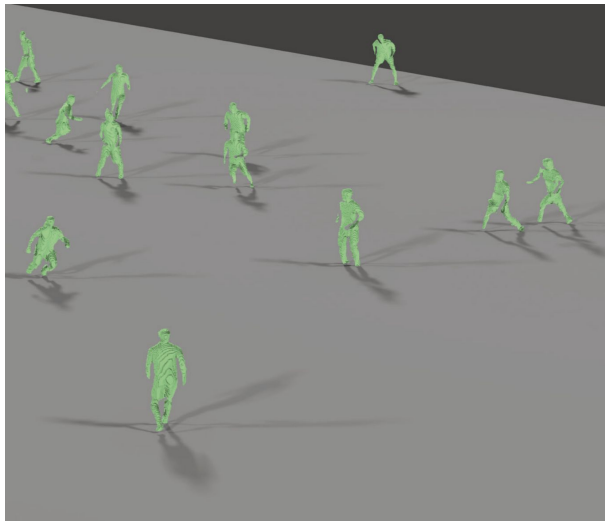
Goal: Develop a robust football player 3D body pose estimation method based on multiple-view soccer streams.

Description:

Intelligent sports video analysis systems have many commercial applications. Recently, with the emergence of 2D body pose estimation algorithms, detailed 3D pose estimation is promising.
The goal of this project is to leverage SOTA multiple person 2D pose estimation algorithms [1] to deliver a robust 3D pose estimation framework based on multiple-view soccer streams. Detailed tasks are:

i: adapt SOTA multiple person 2D pose estimation algorithms for football case.
Ii: triangulate the 2D detection into 3D
iii: in order to evaluate the performance, an annotated sub-dataset is necessary. A semi-automatic annotation tool is required to efficiently conduct this task.

[1] Cao, Zhe, et al. "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields." *IEEE transactions on pattern analysis and machine intelligence* 43.1 (2019): 172-186.



Requirements / Tools:

Python, PyTorch, C++

Supervisor:

Martin Oswald <martin.oswald@inf.ethz.ch>
Jie Song <jsong@inf.ethz.ch>

Recovering 3D player-ball interactions from single images (FIFA)

Goal: Given an image containing a player and the football, recover the 3D body mesh, the 3D football mesh, and their 3D relation.

Description:

The core of a soccer game is player-football interaction. Recovering 3D player-football interactions is essential to understand/predict the player's behavior, and have large potentials for commercial applications, like automatic offside detection.

(1) **Data preparation:** From high-resolution videos provided by FIFA, use algorithms to select frames that contain single player-football interaction. Or, such single player-football interaction should be the primal content.

(2) **Baseline implementation:** There are many methods on 3D body mesh recovery from a single image [1,2]. Also, depending on the football size and how it is occluded by the human body, recovering the football 3D position relative to the player is possible. The work of [3] jointly estimate the object and the person. Then, detect whether the player touches the ball.

(3) **New method based on weak supervision:** In training data, you can have the ground truth label whether the player contacts the ball or not. Use this information to improve state-of-the-art method.

[1] SMPLX https://smpl-x.is.tue.mpg.de/

[2] https://ait.ethz.ch/projects/2020/learned-body-fitting/

[3] https://github.com/facebookresearch/phosa

Requirements / Tools:

Python, PyTorch, C++

Supervisor:

Yan Zhang <yan.zhang@inf.ethz.ch>
Marko Mihajlovic <marko.mihajlovic@inf.ethz.ch>
Jie Song <jsong@inf.ethz.ch>

Fracture Surgery Assistance with HoloLens 2

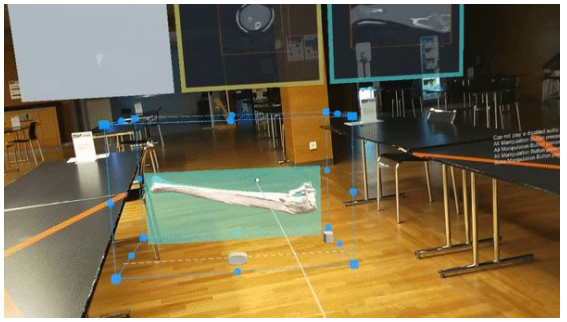
Goal: Extend the existing open-source Hololens App [1] for fracture surgery assistance with novel features.

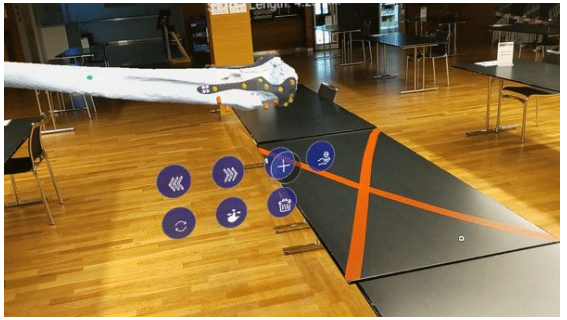
Description:

The goal of this project is to use HoloLens 2 during implant surgeries of complex fractures. The position and orientation of fractured bone pieces is usually not well visible during the surgery and surgeons have to remember their position from the scan inspection prior to the surgery. As a starting point novel app features include:

- real-time 3D collision testing among bone fragments, screws and metal plates during manipulation
- quantitative analysis and visualization of distances, angles among bone fragments, as well as among placed screws (to ensure minimal distances and sufficient bone support)
- intuitive visualization of computed measures and extensions of the GUI

The app development is a creative process with feedback from real surgents in which students are encouraged to contribute their own ideas and additional useful features.





[1] Open-source fracture surgery Hololens2 App: https://github.com/daniCh8/mixed-reality-surgery-assistance-2020
[2] Intro-Video: https://www.youtube.com/watch?v=7BbkqDFOB-g

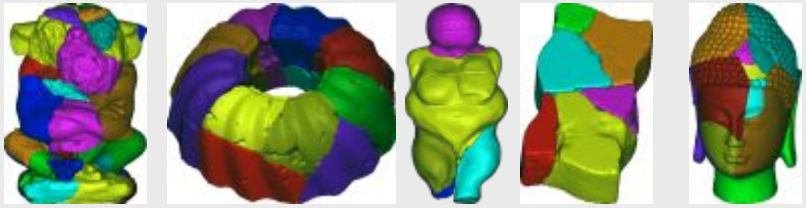
Requirements / Tools:

Unity3D, C#

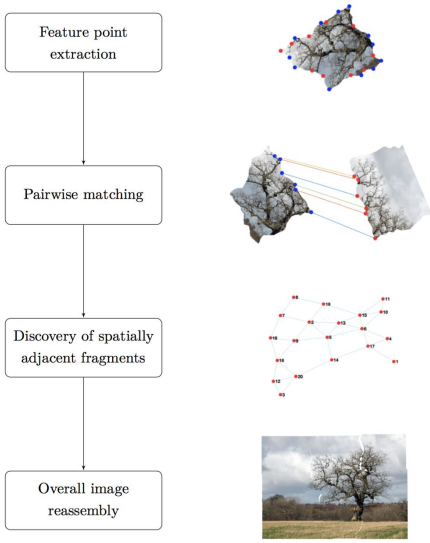
Supervisor:

Martin Oswald <martin.oswald@inf.ethz.ch>
Dr. Sc. Thomas Zumbrunn <tz@ethz.ch>

3D Feature Point Learning for Fractured Object Reassembly



Goal: Create and implement a neural network for learning 3D feature descriptors for efficient matching of fractured geometry pieces.



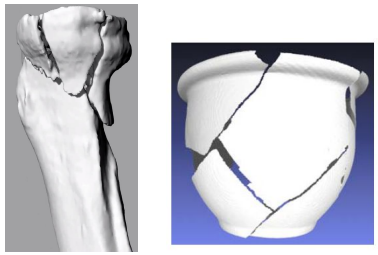
Description:

Fracture reassembly is challenging. The robust and efficient registration of geometric pieces arises in many applications domains, e.g. reconstruction of fractured historic objects like vases, fresces; re-alignment of bone fractures in medical applications; forensics recovering shredded documents; puzzle solving, etc. The matching is done within an 2D/3D optimization framework that was already created and implemented in a past Master thesis (yet unpublished). To demonstrate the full viability of the approach we need show experiments with 3D reassembly problems. Particular tasks:

- implement a neural network for learning 3D feature descriptors suitable for efficient matching of fractured elements (using geometric features, and maybe textures)
- built a synthetic dataset of 3D fractured objects, e.g. using Blender's cell fracture tool [2] and scriptable possibility to add various noise levels on the fractured surface
- Test the learned features with the existing matching algorithm (Matlab)

The goal of this project is a paper submission to a major computer vision conference.

[1] K. Zhang, W. Yu, M. Manhein, W. Waggenspack, X. Li. 3D Fragment Reassembly using Integrated Template Guidance and Fracture-Region Matching, ICCV 2015
[2] Blender Cell facturing tool https://www.youtube.com/watch?v=T2nsntEzlAw



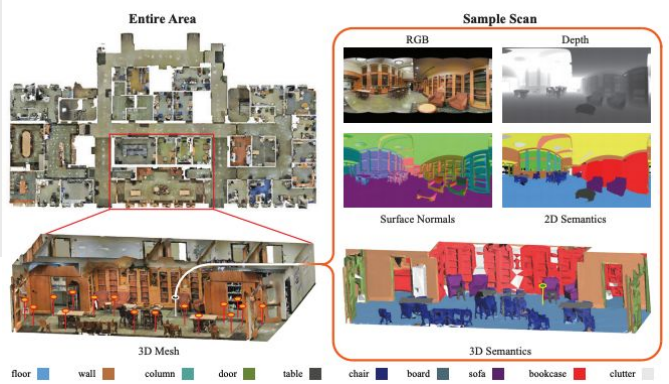
Requirements / Tools:

Python, PyTorch, Blender, Matlab

Supervisor:

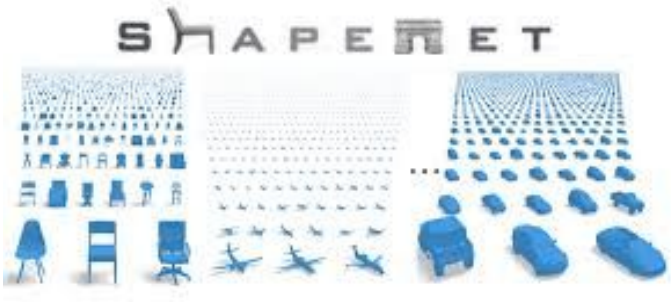
Martin Oswald <martin.oswald@inf.ethz.ch>
Danda Pani Paudel <paudel@vision.ee.ethz.ch>

Real2CAD: Shape Matching of Real 3D Object Data to Synthetic 3D CADs



Goal: Given 3D object data from a real-world (2D3DS [1]), find the closest matching synthetic object given a CAD database (ShapeNet [2]).

Description:



In this project, the team is expected to develop an algorithm that takes as an input a 3D object instance from a real-world dataset (2D3DS [1]) and find the closest matching 3D CAD shape to the input from the ShapeNet database. Matching is evaluated based on the minimum L2 loss based on the entire 3D geometry, as well as that of the part components (using the PartNet [3] database). This is a very interesting problem, given the domain gap between the real and synthetic data. For example, the real data are noisy, have missing parts, etc. The project will begin with the object class of 'chair' and will expand to more object classes as time allows. However, the developed algorithm should be general and not specific to any class.

[1] Armeni, Iro, Sasha Sax, Amir R. Zamir, and Silvio Savarese. "Joint 2d-3d-semantic data for indoor scene understanding." arXiv preprint arXiv:1702.01105 (2017).
[2] Chang, Angel X., Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese et al. "Shapenet: An information-rich 3d model repository." arXiv preprint arXiv:1512.03012 (2015).
[3] Mo, Kaichun, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. "Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 909-918. 2019.

Requirements / Tools:

Python, Pytorch

Supervisor:

Iro Armeni <iro.armeni@inf.ethz.ch>

Next Best View for 3D Reconstruction in noisy and cluttered real-world scenes

Goal: Given an identified object instance in a real-world 3D scene, find the next best view to reconstruct the object in 3D with high fidelity.

Description:

Imagine an agent navigating in an indoor cluttered scene [1] which task is to reconstruct a specific object. Once the object is identified (not part of this project), the agent needs to find the next best view that will provide the most information gain for reaching its goal, a high fidelity 3D reconstruction of the object. The next best view should allow the agent to complete the task in the minimum number of steps required to accomplish the task. The cluttered environment does not allow always a complete viewing of the object. The camera path has certain constraints, e.g., in terms of rotation, simulating a plausible scenario of a camera mounted on top of an agent (the agent potentially can control the camera, something that we can explore in the project). Last, if time allows, the project can increase in complexity if one considers more than one object instances that the agent needs to capture and that are at the vicinity of each other - in this case, the next best view problem has to satisfy information gain for all objects. The project will use the Gibson Environment [2] to simulate the task.

[1] Armeni, Iro, Sasha Sax, Amir R. Zamir, and Silvio Savarese. "Joint 2d-3d-semantic data for indoor scene understanding." arXiv preprint arXiv:1702.01105 (2017).
[2] Xia, Fei, Amir R. Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. "Gibson env: Real-world perception for embodied agents." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9068-9079. 2018.

Requirements / Tools:

Python, Pytorch, RL (optional)

Supervisor:

Iro Armeni <iro.armeni@inf.ethz.ch>

Spatiotemporal 3D Point Cloud registration

Goal: Given 3D point clouds of the same scene over time, that contain dynamic and topological changes, register them with one another.

Description:

New 3D sensor advances allows us to collect 3D data effortlessly and ubiquitously, giving us a timeline of how spaces change over time. Before we begin any analysis on this data, the first step is to register these spaces in the same coordinate system, so that the 3D data of each point in time overlap spatially. This is a difficult problem is one accounts for global registration distortions in each point cloud, as well as for the changes that happen among different points in time (non-static environment). One particularly challenging environment is that of construction sites, where the changes can be drastic in the appearance, geometry, and topology of present objects. The project will focus on such data, of buildings under construction.

Requirements / Tools:

Python

Supervisor:

Iro Armeni <iro.armeni@inf.ethz.ch>

VoroCrust: Polyhedral Mesh Generation

Goal:

Implement the VoroCrust algorithm

Description:

The goal of this project is to implement the VoroCrust algorithm [1] for generation of polyhedral meshes (Figure 1). It works on watertight piecewise-linear complex surfaces (triangle meshes) and covers them through a union of balls (Figure 2, left). The intersection of balls are used to create Voronoi seeds in order to isolate the boundary (Figure 2, middle). Finally, the interior is seeded to acquire a polyhedral decomposition of the mesh (Figure 3, right). The implementation should be done in PyTorch/Python. We're interested in using the generated voronoi seeds as ground truth training data for new surface reconstruction algorithms.

[1] Abdelkader, Ahmed, et al. **"VoroCrust: Voronoi meshing without clipping."** ACM Transactions on Graphics (TOG) 39.3 (2020): 1-16.

[2] Abdelkader, Ahmed, et al. **"VoroCrust Illustrated: Theory and Challenges."** No. SAND2018-3815C. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2018.

[3] https://www.youtube.com/watch?v=PqvTZnekZlY&feature=emb_logo

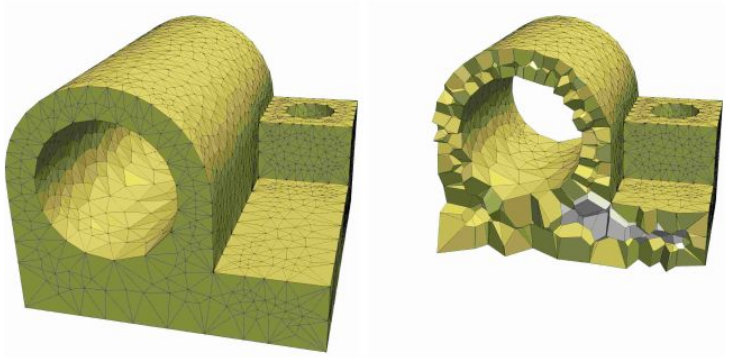


Figure 1: From [1]

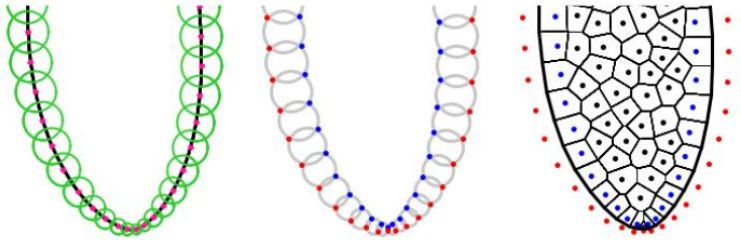


Figure 2: From [1]

Requirements / Tools:

Programming: Python and PyTorch

Supervisor:

Sandro Lombardi <sandro.lombardi@inf.ethz.ch>

DynamicFusion: Reconstruction and Tracking of Non-rigid Scenes in Real-Time

Goal:

Implement DynamicFusion

Description:

Figure 1: From [1]

DynamicFusion is a dense SLAM system capable of reconstructing non-rigidly deforming scenes in real-time [1]. DynamicFusion uses a deformation graph to model the non-rigid deformation in a scene. New depth frames are continuously fused into a reference frame by warping the reference frame to the current time step with the last known deformation.

We're interested in DynamicFusion as a baseline method. The implementation should be done in PyTorch/Python. Several pieces of code can be provided (marching cubes, depth unprojection, some parts for the deformation model, etc). Real-time performance is not a requirement.

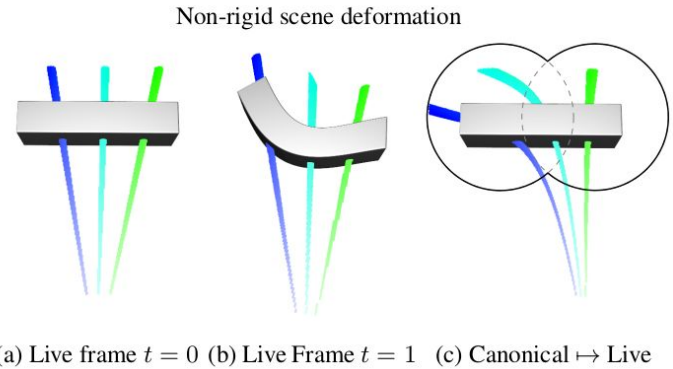


Figure 2: From [1]

[1] Newcombe, Richard A., Dieter Fox, and Steven M. Seitz. **"Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time."** Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

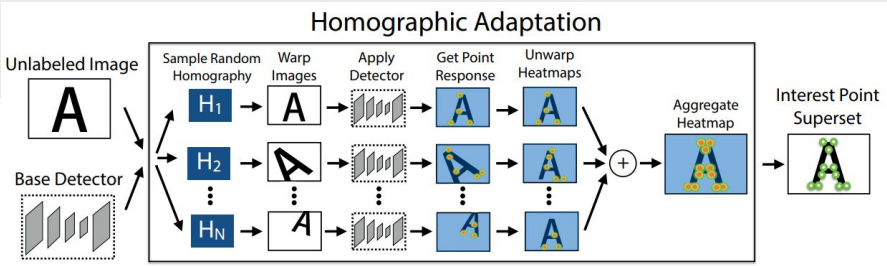
Requirements / Tools:

Programming: Python and PyTorch

Supervisor:

Sandro Lombardi <sandro.lombardi@inf.ethz.ch>

Viewpoint Adaptation in a Synthetic Environment



Goal: Implement viewpoint adaptation for training of local features in a synthetic environment

Description:

In SuperPoint [1], the authors proposed a technique called homographic adaptation for self-supervised training of local features. The main issues with the homographic assumption is that real world scenes cannot be supposed to be planar (especially indoors) and occlusions are not taken into account.

The plan is to extend this notion to viewpoint adaptation - i.e., instead of using synthetic warps, one can use renders from different viewpoints and similarly aggregate the detection maps.

For this purpose, we take advantage of the recently released photorealistic HyperSim [2] dataset which comes with ground-truth camera parameters and depth. This project can either build upon existing implementations or write a novel training pipeline from scratch.

[1] SuperPoint: Self-Supervised Interest Point Detection and Description, DeTone et al., CVPRW 2018

[2] Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding, Roberts and Paczan, arXiv 2020

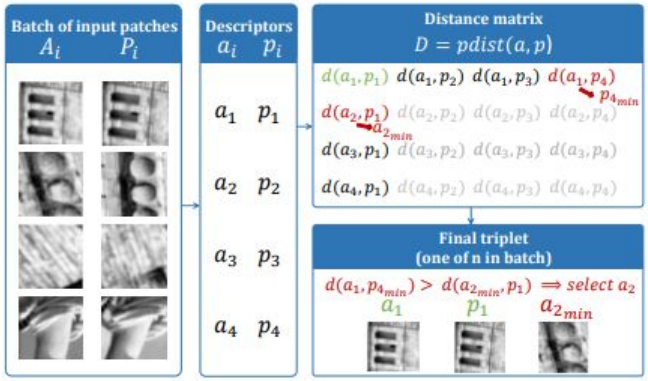
Requirements / Tools:

Python, PyTorch

Supervisor:

Mihai Dusmanu <mihai.dusmanu@inf.ethz.ch>

Binary HardNet



Goal: Experiment with different losses for binary patch description

Description:

One main advantage of binary local descriptors is their efficient storage and matching using simple bitwise operation - popcnt(a land b). However, most modern descriptors achieving state of the art performance are floating point.

The goal of this project is three-fold:
- re-implement a simplified version of the HardNet [1] training pipeline which uses hardest in batch negative mining
- reproduce the results reported in the paper
- experiment with different baseline / state-of-the-art losses for binary descriptors [2]

[1] Working hard to know your neighbor's margins: Local descriptor learning loss, Mischuk et al., NeurIPS 2017

[2] Learning Deep Binary Descriptor With Multi-Quantization, Duan et al., CVPR 2017

Requirements / Tools:
Python, PyTorch

Supervisor:

Mihai Dusmanu <mihai.dusmanu@inf.ethz.ch>

On the Robustness of Local Feature Inversion Techniques

(b) Projected 3D points (c) Synthesized Image (d) Original Image

Goal: Study the robustness of local feature inversion networks under different conditions

Description:

Local feature inversion is a privacy attack that aims to reconstruct images from their local features (e.g., SIFT) placed at the closest grid location. One such recent work [1] uses a coarse reconstruction U-Net trained using perceptive and pixel reconstruction losses followed by a fine reconstruction U-Net trained in an adversarial fashion.

There are several directions for this project:
- Evaluate the approach with learned features (e.g., SuperPoint [2], R2D2 [3])
- Evaluate the generalizability to novel scenes / data required for training
- Evaluate the robustness to a subsampling in terms of keypoints / feature dimension
- Improve the architecture / training loss

[1] Revealing scenes by inverting structure from motion reconstructions, Pittaluga et al., CVPR 2019

[2] SuperPoint: Self-Supervised Interest Point Detection and Description, DeTone et al., CVPRW 2018

[3] R2D2: Repeatable and Reliable Detector and Descriptor, Revaud et al, NeurIPS 2019

Requirements / Tools:

Python, PyTorch

Supervisor:

Mihai Dusmanu <mihai.dusmanu@inf.ethz.ch>

AR Tourist Guide for Viewing Platforms

Goal: Develop a AR application for smartphones that can be used on Polyterrasse, Rigiblick and Uetliberg.

Description:

Many touristic viewing platforms have a panel that indicates landmarks and sights for tourists and also provides tourists with more information about specific sights. However, these panels are usually crowded, which is especially in times of social distancing not ideal. Therefore, the goal of this project is to develop an AR tourist guide for visual platforms (Polyterrasse/Rigiblick/Uetliberg). It should detect predefined sights (e.g. Grossmünster, Urania-Sternwarte) and landmarks (Lake of Zurich/Uetliberg) and potentially provide the user with some more information. Ideally, it works at different times of the day.

Requirements / Tools:
Android Studio, Python, C++

Supervisor:
Silvan Weder <silvan.weder@inf.ethz.ch>

Stable View Synthesis

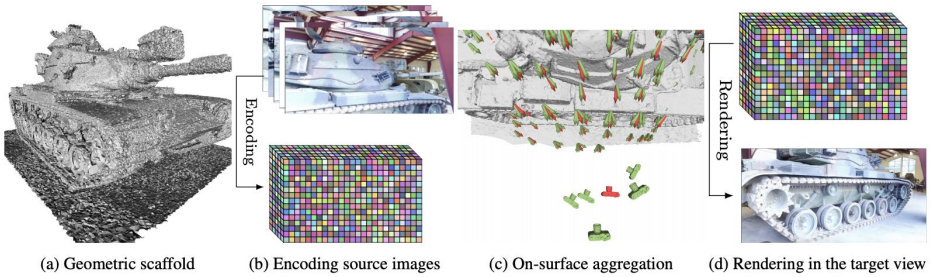
Goal: Implement and reproduce stable view synthesis on challenging scenes.

Description:

Novel view synthesis using neural networks has emerged as a very active field of research in recent years. It allows to render novel views from a scene that has been captured with multiple images.

The goal of this project is to implement stable view synthesis and investigate scenarios, where it works well and where it fails, beyond the results shown in the paper.

[1] Riegler, G., & Koltun, V. (2020). *Stable View Synthesis*.



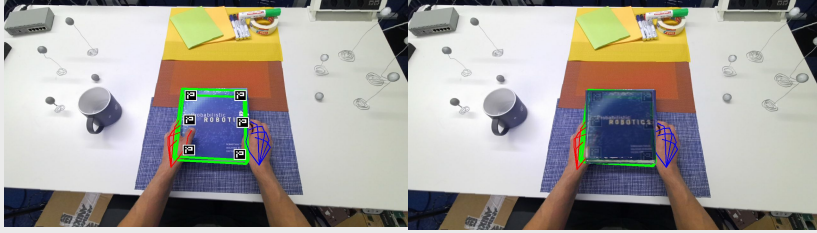
Requirements / Tools:

Python, PyTorch

Supervisor:

Silvan Weder <silvan.weder@inf.ethz.ch>

AR Dataset for Egocentric Action Recognition



Goal: Collect an action recognition dataset that contains head & hands & object pose and eye gaze using HoloLens2.

Description:

Understanding the actions of people in an egocentric view is one of the most important problems for mixed reality and computer vision. However, from egocentric viewpoints, recognizing actions is challenging due to fast camera motion, blur, (self-)occlusion and cluttered backgrounds. Using hand keypoints and object poses for action recognition is one way to deal with those problems effectively [2].

HoloLens2 research mode [1] provides hands & head pose and eye gaze. However, the research mode can't capture object pose which is regarded as one of the most important components to understand action recognition in an egocentric view [2]. Therefore, a robust way to obtain object poses in augmented reality is suggested in the following procedures.

(1) The object pose is calculated using visible markers such as infrared (IR) or printed ones.

(2) Markers, which can cause an overfitting problem during training, will be hidden from the users' viewpoint by projecting 3D meshes on an object. Students need to build an app for this on HoloLens2.

(3) Students will collect a dataset with actions using the app. Users will see a projected object instead of the original object (3d printed, only shape) with markers.

(4) The state-of-the-arts action recognition algorithms [3,4] will be used to verify the dataset.

[1] Ungureanu, Dorin, et al. "HoloLens 2 Research Mode as a Tool for Computer Vision Research." arXiv preprint arXiv:2008.11239 (2020).
[2] Tekin, Bugra, Federica Bogo, and Marc Pollefeys. "H+ o: Unified egocentric recognition of 3d hand-object poses and interactions." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.
[3] Feichtenhofer, Christoph, et al. "Slowfast networks for video recognition." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.
[4] Liu, Ziyu, et al. "Disentangling and unifying graph convolutions for skeleton-based action recognition." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

Requirements / Tools:

Python, C#, Unity3D, HoloLens

Supervisor:

Taein Kwon <taein.kwon@inf.ethz.ch>

Catadioptric stereo

Image: Donald Simanek

Goal: Implementing a depth estimation algorithm for images of a monocular camera observing a mirror.

Description:

Modern smartphones often come with multiple cameras to help with depth estimation. This is later used for 3D reconstruction, but also to foreground / background separation in images.
However, a much simpler setup has been known to work well for a long time [1]. Using only a single camera, depth of a scene can be estimated if this scene is observed both directly and as reflection in a mirror in the same image.
This project involves building a physical camera setup with a mirror and implementing the necessary algorithms for calibration and depth estimation.

[1] Gluckman and Nayar, Catadioptric Stereo Using Planar Mirrors, IJCV, 2001

Requirements / Tools:

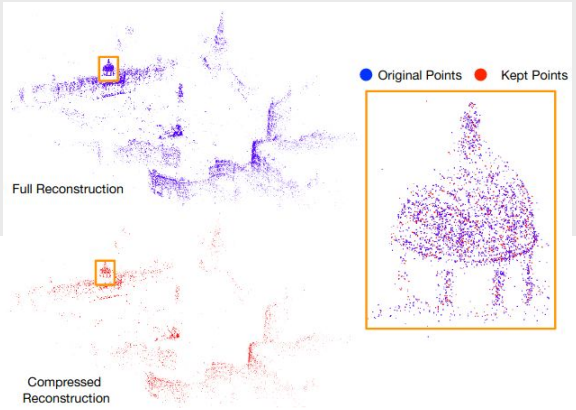
Knowledge in classical computer vision / geometry

C++

Supervisor:

Daniel Thul <daniel.thul@inf.ethz.ch>
Marcel Geppert <marcel.geppert@inf.ethz.ch>

Map compression for visual localization



Goal: Implementing and evaluating a novel map compression algorithm

Description:

One big problem for large scale localization is the sheer amount of data that need to be processed. Typical 3D maps can easily contain millions of points that need to be stored and considered during localization.
Compressing these maps can not only reduce the data that need to be handled, but even improve the localization robustness by removing ambiguous parts of the scene.
The goal of this project is to implement a previously presented algorithm for map compression [1] and to evaluate the changes by adapting an existing visual localization pipeline [2].

[1] Efficient Scene Compression for Visual-based Localization, Mera-Trujillo et al., 3DV 2020

[2] http://www.graphics.rwth-aachen.de/software/image-localization/

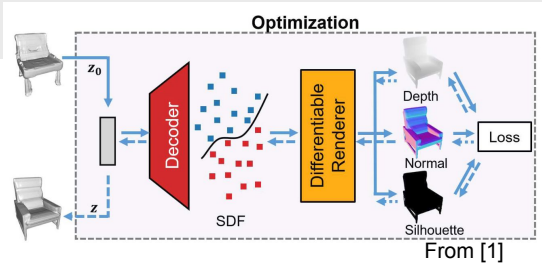
Requirements / Tools:

C++

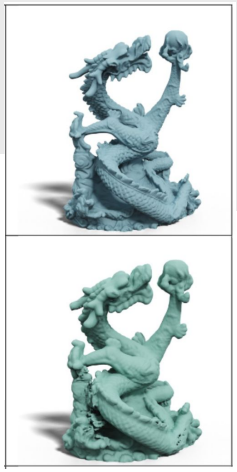
Supervisor:

Marcel Geppert <marcel.geppert@inf.ethz.ch>

Exploring Differentiable Rendering for Neural Implicit Functions



From [1]


From [2]

Goal: Comparison and extension of sdf-based differentiable renderer.

Description:

Differentiable Rendering provides a very useful tool for self-supervised learning of 3D shapes and texture of objects from images. Recent rendering modules were developed for different types of 3D representations, like meshes, voxel grids, or neural implicit functions. In this work, we focus on neural implicit functions which benefit from constant memory usage for arbitrary resolution and are not restricted to a fixed topology.

The project consists of the following tasks:
1. comparison of sota renderer DIST[1] and SDFDiff [2]
2. adaptation of renderer of choice for the task of single-view 3D shape and texture reconstruction of varying objects from the same category
3. further extension e.g. by adding an explicit lightning model, object pose prediction, generation of novel scenes

[1] 'DIST: Rendering Deep Implicit Signed Distance Function with Differentiable Sphere Tracing', Liu et al., CVPR'20
[2] 'SDFDiff: Differentiable Rendering of Signed Distance Fields for 3D Shape Optimization', Jiang et al., CVPR'20

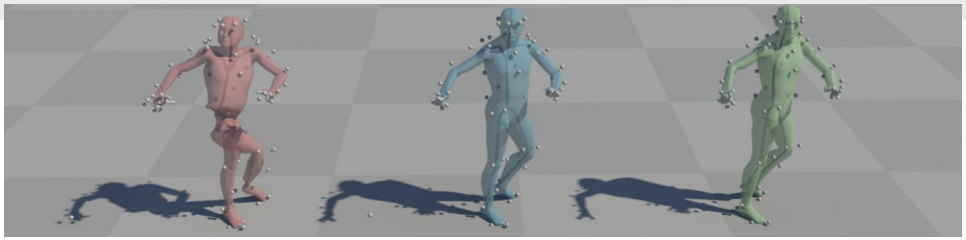
Requirements / Tools:

Python, Pytorch

Supervisor:

Cathrin Elich <cathrin.elich@inf.ethz.ch>

Regressing 3D motions of body mesh from sparse markers



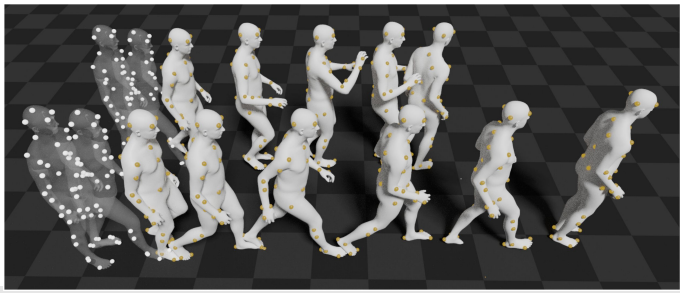
Goal: Compare existing methods and design new architectures to regress SMPLX parameters from sparse surface markers.

Description:

Marker-based human motion capture (a.k.a. mocap) is to obtain high-quality data, and is usually the very first step in many computer vision and computer graphics tasks. In the modern mocap system, what we get is actually a time sequence of locations of sparse markers on the body surface. They are normally corrupted, noisy, and frequently occluded by different human body parts. Therefore, they cannot be directly used, but need to fit body skeletons or parametric body mesh models (more popular) in a highly precise manner. However, current state-of-the-art methods are very time consuming, and require massive manual work.

In this project, we aim to propose a neural network to regress the body mesh parameters from corrupted body surface markers. A well-designed neural network can significantly speed up and automatize this job, while retaining the high quality. Specifically, our goals are as follows:

1. Get familiar with the body mesh model SMPL-X[1] and how the large-scale AMASS[2] is created.
2. Based on AMASS, we get virtual markers from the body mesh, and use them as training/testing data.
3. Compare existing methods and baselines.
4. Propose your own network to outperform all existing methods.



[1] SMPLX:. https://ps.is.mpg.de/publications/smplex-2019
[2] AMASS and Mosh++: https://amass.is.tue.mpg.de/
[3] MOJO: https://yz-cnsdqz.github.io/MOJO/MOJO.html
[4] Holden, Robust Solving of Optical Motion Capture Data by Denoising, TOG 2018

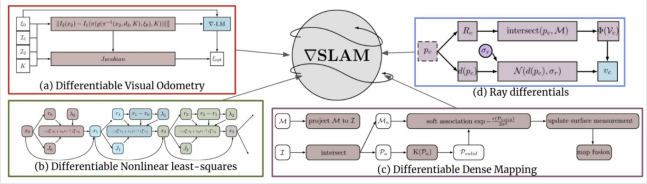
Requirements / Tools:

Python, Pytorch, Open3d, Pytorch3D, blender

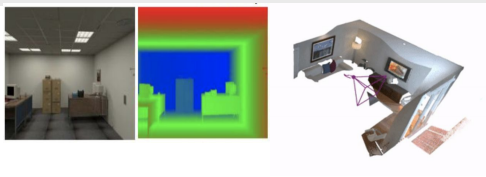
Supervisor:

Yan Zhang <yan.zhang@inf.ethz.ch>
Siwei Zhang <siwei.zhang@inf.ethz.ch>

End-2-end self-supervised SLAM



Goal: Build an online self-supervised SLAM pipeline for real-time dense reconstruction



Description:

Dense monocular SLAM is a challenging task since 3D structures reconstructed from monocular images are often sparse and It is not easy to recover reliable 3D information for non-textured regions in real-time. The use of a depth prediction network within a SLAM pipeline has been proposed to improve dense reconstructions [1], however, deep learning models suffer from drops in accuracy on scenes not similar to the training ones (domain shifts).
The goal of this project is to use online adaptation to overcome this generalization issue and build a fully differentiable SLAM pipeline that can be optimized end-2-end. We propose to combine the SLAM framework of [2] with the self-supervised depth prediction network in [3] to optimize a SLAM pipeline for real-time dense reconstruction on totally unseen scenes.

[1] CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction. Tateno et.al. (2017)
[2] GradSLAM: automagically differentiable SLAM. Murthy et al. (2020)
[3] Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos. Casser et al. (2019)

Requirements / Tools:

Python, DL frameworks (Tensorflow or Pytorch)

Supervisor:

Keisuke Tateno <ktateno@google.com>
Alessio Tonioni <alessiot@google.com>
Federico Tombari <tombari@google.com>
Paul-Edouard Sarlin / psarlin.com / psarlin@ethz.ch

Patient tracking for HoloLens

Goal: Develop and test the best method to track objects accurately using the HoloLens

Description:

Augmedit is a start-up that aims to improve the insight of doctors and patients by transforming 2D imaging into 3D holograms. We have created a workflow that can be used to automatically transform MRI scans of patients with brain tumours into 3D holograms for the HoloLens 2 *(fig 1)*. These holograms can be used by doctors to prepare or plan their surgical procedure. The spatial awareness of the HoloLens allow it's user to place 3D holograms anywhere in the room and let the user appreciate or manipulate these models. However, it remains challenging how these 3D holograms can be 'fused' with the real patients. Accurate tracking of patients is essential for the next step: to use 3D holograms for guidance during surgery *(fig 2)* [1]. Possible solutions that can be used include 2D marker tracking *(fig 4)* or use the various sensors of the HoloLens to develop and implement even better tracking methods *(fig 3)*.

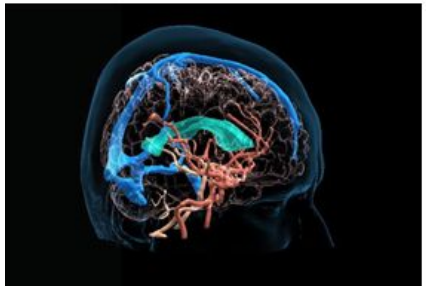


Figure 1: 3D hologram created with Augmedit software

Figure 2: The aim is to accurately match holograms on patients

Figure 3: various sensor streams available in Research Mode that could be used to track objects (image Microsoft)

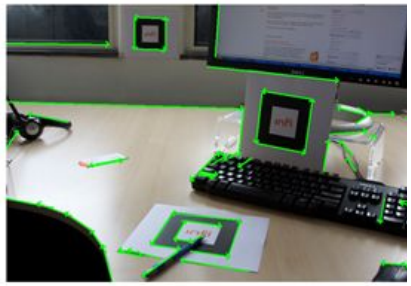


Figure 4: Example of image tracking that could be used by the RGB camera

[1] T. Fick, J. van Doormaal, E. Hoving, L. Regli, T van Doormaal - Holographic patient tracking after bed movement for augmented reality neuronavigation using a head-mounted display. Acta Neurochir (Wien). 2021 Jan 29. doi: 10.1007/s00701-021-04707-4.

Requirements / Tools:

Unity 3D, C#

Supervisor:

Jene Meulstee <jene@augmedit.com>
Tristan van Doormaal <tristan@augmedit.com>
Jonas Hein <jonas.hein@inf.ethz.ch>