Popular Summary

As humans, we are capable of perceiving all three spatial dimensions (3D) of our surroundings through our eyes and motion alone. Even though our eyes – like cameras – only capture a flat projection of the actual world, we have learned how to think in 3D. For example, by moving around in a house we learn the layout and size of the rooms; by walking around a block in the city we can build a sense of direction; through experience we can estimate the height of a cabinet without relying on a measuring tape.

Can we teach a computer to do this? This thesis studies a set of related questions in the field of *Computer Vision*. As tools we use geometrical mathematical models and *neural networks*, which are complex mathematical models inspired by the human brain models that require a large amount of data to learn from. Examples of questions are: Can we train neural networks to reason about hidden spaces indoors, as in Figure 1? Can we train neural networks to figure out the room layout in terms of floors, walls and ceilings from a single photo? How can we efficiently harness motion to estimate the 3D world from photos taken at different locations?

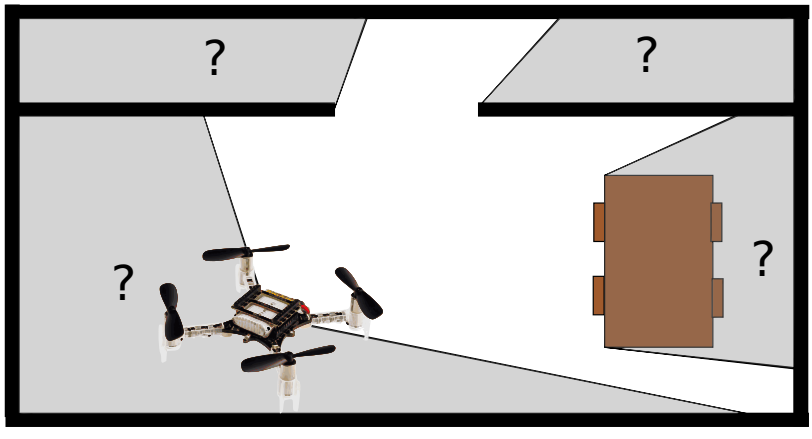

Figure 1: Quadcopter in a room seen from above. It is trying to guess what may be hidden from view behind the walls and table.

The brain's ability for abstract reasoning is one of many keys to our spatial awareness. To orient ourselves we may make note of key objects – such as houses, signs, trees etc. – that we use to track our current motion and help us find our way when we revisit the location. Within Computer Vision, the problem of understanding the 3D environment from photos is known as *Structure from Motion* (SfM). The first step is typically to identify good key objects or pixels in the photo that we can use to relate to the other photos.

The standard solution is to use *keypoints* in the image, with a colour intensity variation signature called a *descriptor*. The descriptor is used to find tentative matching points in other

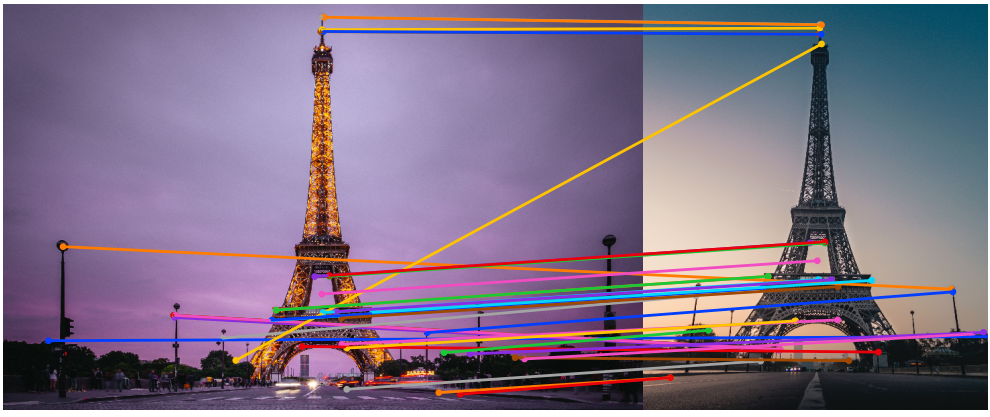

Figure 2: Two photos taken of the Eiffel Tower at different times from different locations. Keypoints (SIFT) are extracted from each image and matching by appearance is attempted. We see that while some matches are accurate, a large part of the tower has no matches and some are incorrect.
Image credit: Denys Nevozhai and Gautier Salles from `unsplash.com`.

images. While often sufficient, the matching methods struggles with repeated patterns like brick walls or homogeneous surfaces such as white painted walls. Changes over time are also difficult – for example, in Sweden a tree in winter looks very different to a tree in summer. In Figure 2 we see keypoints extracted from two images of the Eiffel Tower, but at different times of the day. There are only some parts of the tower that match correctly under these differing circumstances.

To alleviate these problems this thesis shows how e.g. trees and poles can be modelled as parallel cylinders to improve robustness and efficiency. Additionally it studies indoor scenes and how lines and polygons may be detected to represent the room layout in terms of floors, walls, ceilings, windows and doors as illustrated in Figure 3. This higher level of abstraction regarding key objects also improves robustness and is essential to reach human levels of 3D reasoning.

After key points and objects have been identified, it is time to estimate the pose (position and orientation) of the objects and of the cameras used to capture the photos. To find matching points or objects between image pairs, a robust matching method is needed that is capable of ignoring incorrect matches. A common method is the *Random Sample Consensus* (RANSAC) algorithm, which repeatedly solves a *minimal problem* for an alternating small set of random points. It is important to develop fast solvers for these minimal problems, as this makes it possible to match images in real time or in large scale. For example, estimating relative position and orientation between two calibrated cameras requires at least 5 corresponding points in each image. The solvers for this problem developed by the research community requires only microseconds to execute. Therefore, it is possible to perform thousands of RANSAC iterations in only milliseconds to find the best possible set of points. In this work, fast solvers are presented for both matching of parallel cylinders

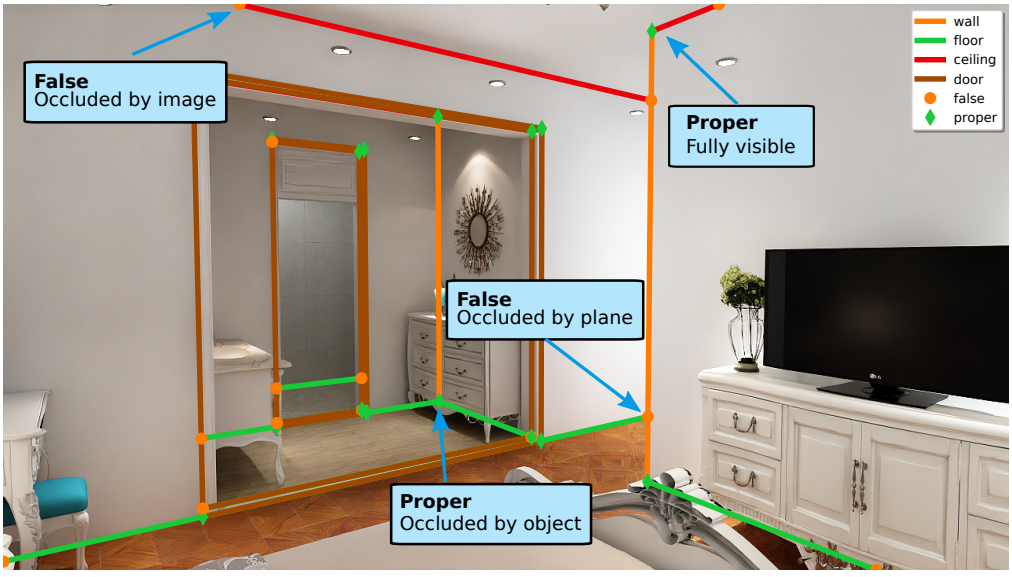

Figure 3: Paper II propose a wireframe representation for room layouts. The lines and points may be used for SfM and are stable since they depend on the room structure rather than appearance of furniture and wallpapers.

and flexible matching of points.

After the matching is done it is time to optimize all positions and camera poses to minimize the error in the map. This is called *bundle adjustment* and is an iterative process to reduce the *reprojection error*, which is the error obtained by projecting a key point or key object from the 3D model back to the photo and comparing the projection with the captured point or object. If using only keypoints, then the finished map is a set of points – also known as point cloud – which represents the surrounding structure.

Performing bundle adjustment on a large-scale problem is difficult since the computations required do not scale proportionally to the number of images. Meaning that while we can process the outside of a building in a few minutes on a laptop it would take days to process a block with the same method. There are of course several ways to make it feasible, most often by splitting the problem into smaller pieces. This thesis studies *map merging*, which aims to as effectively as possible merge two or more point clouds without performing a new bundle adjustment. Methods are developed which optimize the reprojection error and do not require a full overlap of the points. It is useful for reducing computations but still flexible enough to correct shapes of objects if they are incorrect.

In short, this work touches on many aspects of the Structure from Motion pipeline. From detection of lines and polygons in images, to minimal solvers and merging of point clouds. Hopefully, the contributions may be a useful piece in the puzzle to enable autonomous systems and other services to understand our surroundings as we do.