The Role of Object Instance Re-Identification in 3D Object Localization and Semantic 3D Reconstruction

Vaibhav Bansal

Department of Electrical, Electronic and Telecommunications Engineering, and Naval Architecture (DITEN)

Università degli studi di Genova

Supervisor

Alessio Del Bue

In partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computing of the University of Genova

November, 2019

Acknowledgements

I would like to express my gratitude to Dr. Alessio Del Bue, Prof. Mario Marchese, Dr. Stuart James, Dr. Paul Gay, my friends at PAVIS and VGM and lastly my parents for their constant support in this journey.

Abstract

For an autonomous system to completely understand a particular scene, a 3D reconstruction of the world is required which has both the geometric information such as camera pose and semantic information such as the label associated with an object (tree, chair, dog, etc.) mapped within the 3D reconstruction.

In this thesis, we will study the problem of an object-centric 3D reconstruction of a scene in contrast with most of the previous work in the literature which focuses on building a 3D point cloud that has only the structure but lacking any semantic information. We will study how crucial 3D object localization is for this problem and will discuss the limitations faced by the previous related methods. We will present an approach for 3D object localization using only 2D detections observed in multiple views by including 3D object shape priors.

Since our first approach relies on associating 2D detections in multiple views, we will also study an approach to re-identify multiple object instances of an object in rigid scenes and will propose a novel method of joint learning of the foreground and background of an object instance using a triplet-based network in order to identify multiple instances of the same object in multiple views. We will also propose an Augmented Reality-based application using Google's Tango by integrating both the proposed approaches. Finally, we will conclude with some open problems that might benefit from the suggested future work.

Contents

List of Figures

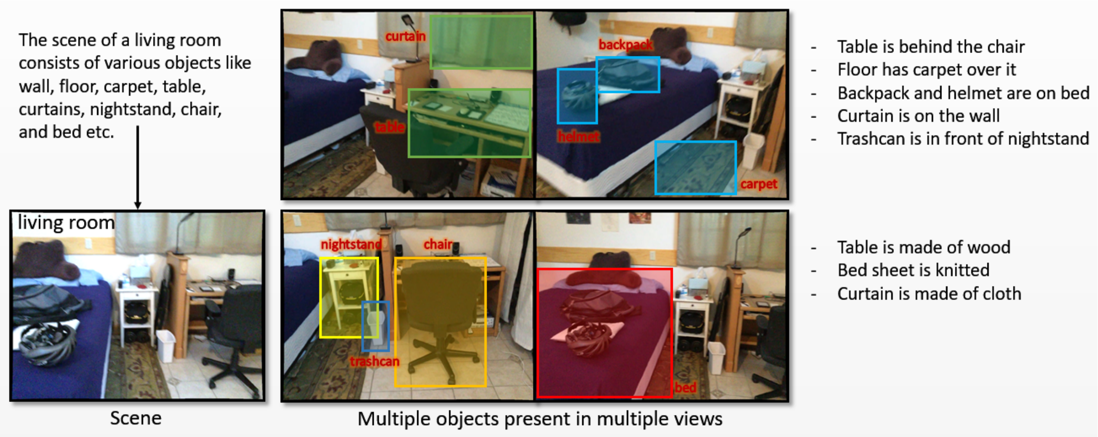
List of Tables

Chapter 1

Introduction

1.1 Motivations - the Human Vision

At the first glance, what we see with our eyes can be considered as mere images made up by the visual light reflected off of the surfaces of the objects that we are looking at. But, how can we perceive more than what is captured in a 2D picture? Certainly, there is more to the human vision than meets the eye. Our eyes work more like a camera that captures an image. Perception really happens in the brain which turns those images into something that we can understand. Our incredibly complex visual system including the two eyes, optic nerves and the brain allow us experience the visual world around us in three-dimension. Our visual system has evolved over millions of years to recognize and understand very accurately and with low latency the complex visual world around us. We are able to perceive, analyze and extract a tremendous amount of semantic and geometric information for an elaborate interpretation of the 3D world surrounding us. In order to interact with the 3D world, we not only need to instantly identify various objects present in a particular scene but also to identify the fine characteristics such as materials, textures, different parts, the surfaces that support them and



The scene of a living room consists of various objects like wall, floor, carpet, table, curtains, nightstand, chair, and bed etc.

- Table is behind the chair
- Floor has carpet over it
- Backpack and helmet are on bed
- Curtain is on the wall
- Trashcan is in front of nightstand

- Table is made of wood
- Bed sheet is knitted
- Curtain is made of cloth

Figure 1.1: The scene of a bedroom with various objects like lamp, chair, bed etc. We detect various objects in the scene and build an understanding of structure and arrangement of the scene.

their relative position and volume or depth in order to manipulate them. For example, in Figure 1.1, we can recognize the scene as a bedroom which has various *objects* like a table, chair, bed and lamp etc. We can also observe various fine details about the objects and their *relation* with other objects like the painting *on* the wall, the carpet *on* the floor, the lamp on top of the *wooden* table, a helmet and a backpack *on top of* the bed, chair *in front of* the wooden table, trashcan in front of the nightstand and even the fact that the bedsheet is *knitted*. If we observe, in inferring all of this information, our visual system localized and recognized all the objects within the scene and their *spatial relationship* with their environment. Thus, for a semantic scene understanding, localizing objects in the scene becomes inevitably important. But, first let us understand what is semantic scene understanding in general.

1.1.1 Understanding a scene

When we described the scene in the above example, we described the different objects in the scene. Does it mean that a scene is simply a collection of different

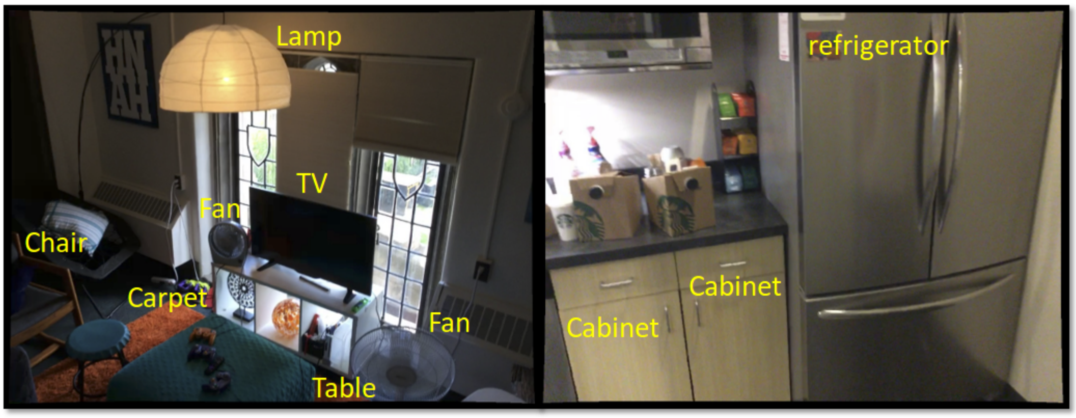


Figure 1.2: Presence of certain objects belonging to specific object categories can provide some context as to which category the scene might belong to. Left: The scene of a living room with objects like lamp, chair, TV and fan etc. Right: Objects like cabinets, drawers and a refrigerator are more common in the kitchen than the living room.

physical objects? A scene, as we humans understand it, is a view of the 3D world that consists of various objects of different shapes and sizes organized in a meaningful way with their real-world functions known to us in a particular environment given a particular context.

It is not just sufficient enough to identify, localize targets and estimate their volumes in a given scene but the context is also crucial for understanding a scene completely. Understanding the context goes beyond merely recognizing objects and their position, it is interpreting how a number of different objects belonging to different object classes interact with each other and/or with their environment. For example, from Figure 1.2 the recognition of some specific objects such as TV, chair and fan and their relationship with the environment may provide enough context for a scene classification task to identify this particular scene to be, at the least, different than the scene of a kitchen where objects like cabinets, sink and a refrigerator would be more common. A combination of objects like a mirror, sink and a toilet seat could certainly help classifying the scene as a bathroom.

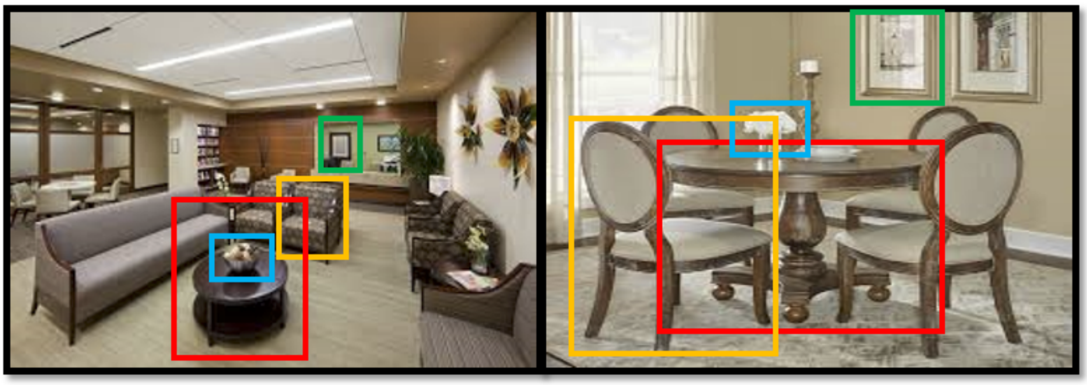


Figure 1.3: Presence of similar objects in two scenes does not mean that the two scenes must be the same. Without context, it is not easy to distinguish between complex scenes. Left: A scene of a waiting area with some chairs or sofa around a round table, a flower pot sitting on the table and some paintings on wall at the far end. Right: A scene of a dining room with similar objects.

Thus, the presence of certain objects in a scene might provide important cues in understanding the type of a scene.

1.1.2 Context is crucial

Even when the objects are correctly recognized and localized in a scene, without the context there is no real understanding. In fact, there is a very high possibility that a scene classification task, for example, would yield inaccurate results if no context is available since the same objects could be found in any other scene. For example, as we can notice from Figure 1.3, it is not easy to distinguish between different complex scenes that have similar objects present. Both the waiting area and the dining room may contain similar objects such as chairs, a round table, a flower pot and some paintings and thus, the presence of similar objects doesn't provide enough discriminative information regarding the category of the two scenes in which they are present. Hence, while recognizing the existence of certain objects in a scene might narrow down the search space for a scene clas-

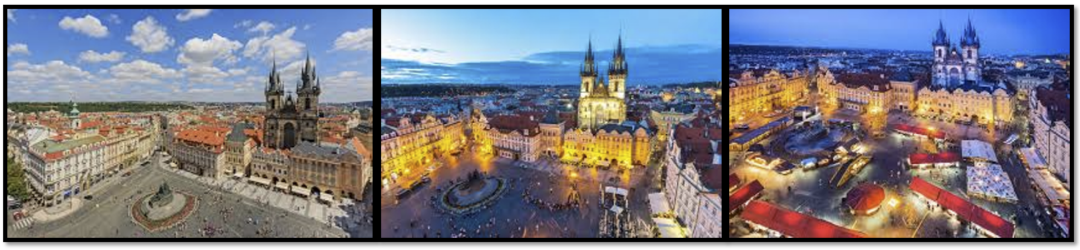


Figure 1.4: Humans can generalize well the already learned concepts by applying them to a different context or domain. For example, a human can easily recognize a place in different images obtained with different lighting conditions which is still very complex to achieve with any autonomous system.

sification task, for instance, there is just not enough evidence to distinguish that particular scene from another when similar objects are detected in both. In such scenarios, some prior knowledge about the different categories of the scenes could provide the context and thus, the discriminative factor between the two scenes. For example, the dining room would have a certain spatial arrangement of chairs around the round table while the waiting area would require a different arrangement. Thus, an *inter-object spatial relationship* can be one of such examples of a very crucial contextual information that can help us distinguish between the given two scenes.

Once the visual concepts and the context about a particular scene are learned, can this understanding be transferable to new variations in the same scenarios previously observed? Humans have a tremendous capability to learn and adapt to new and different situations. We can adapt from scene to scene, given the context, accomplishing all the necessary tasks in every scenario like recognizing targets and estimating their structure and motion. For example, we can recognize the same place in different images captured in different lighting conditions such as during the day and the night as shown in Figure 1.4. Moreover, to perform different tasks efficiently we can adapt and vary our focus on different specific features in an image using the context. For example, to identify a dog in an image

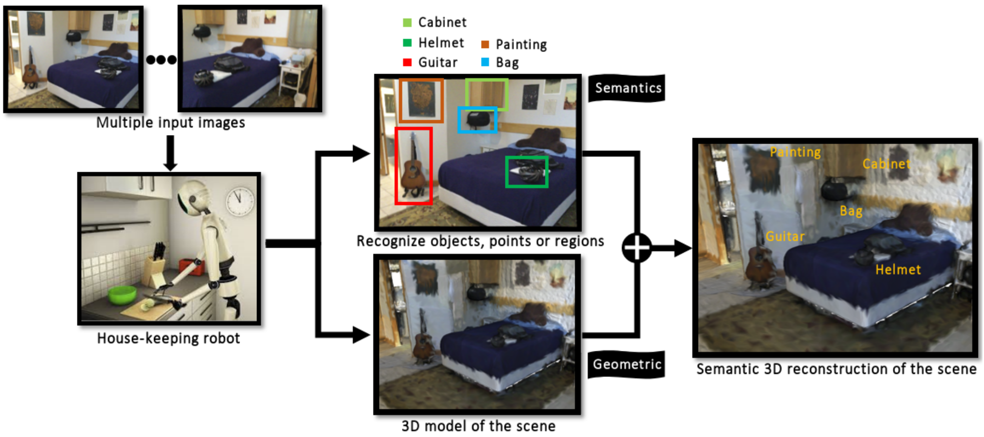


Figure 1.5: Any autonomous system like a house-keeping robot needs to extract both the geometric and semantic information from the images in order to create a meaningful 3D reconstruction of the world.

of a bedroom, only the characteristics related to the dog like shape, color and texture related to the animal are required, the characteristics of the bedroom are not necessary. The same dog would be identified easily in any other environment because of the high capability of the human visual system to adapt and generalize to a new context or domain. On the other hand, to understand the characteristics of the room, we need to pay attention to various objects like the furniture, other common objects and their spatial relationships instead of the dog.

1.2 Autonomous system to understand a scene

So, how to design an autonomous system with a human-like perception? Taking inspiration from the human visual system, an autonomous system with near-human capabilities should be able to identify the visual concepts or cues such as edges, corner points or regions and recognize various objects present in the scene. As illustrated in Figure 1.5, both the geometric information such as camera pose, depth etc. and semantic information such as semantic labels for various regions

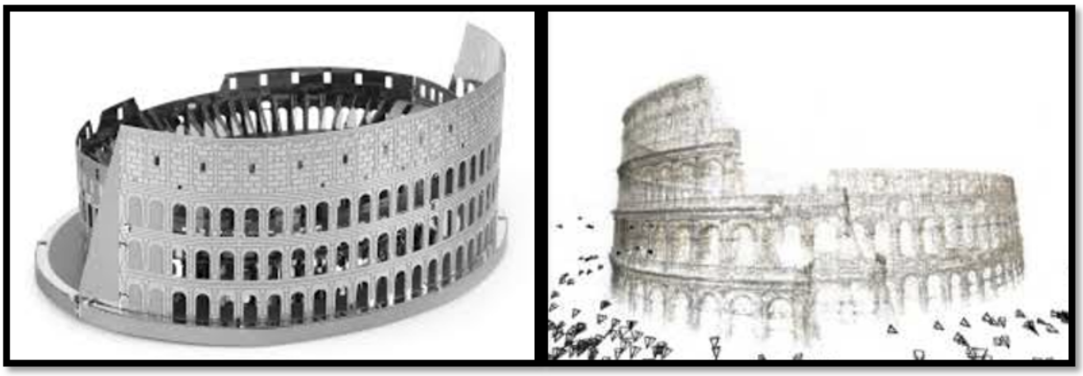


Figure 1.6: An autonomous agent needs to build a digital representation of the 3D world in order to analyze and better understand it, just like humans use physical 3D models of an environment such as a famous monument like the Colosseum.

or objects are all very crucial for interpreting the structure and semantics of the 3D world.

However, designing such an autonomous system has been a continuous challenge till date for all the scientists and engineers working in the field of computer vision for decades. To understand the challenges in semantic scene understanding, it is important to discuss 3D scene reconstruction and eventually, the 3D localization of objects.

For a complete scene understanding, a computer system needs to build a representation of the 3D world. Just like for our better understanding of a particular environment, say a building, we use some physical 3D models for analysis (Figure 1.6), in a similar manner the autonomous agent needs to use a digital 3D model of the environment in order to analyze and understand it. In order to construct such a geometric representation of the world, an autonomous system first needs to scan the real-world to acquire data using specific sensors including regular camera, depth sensors, multi-spectral cameras, laser scanners etc. or sometimes even a combination of any of these sensors depending upon the type of application. The kind of the data acquisition method used shapes how the data is processed by the system in later stages. With the advancement of 3D sensors, many reconstruction techniques work directly with the acquired 3D data. Most popular of them could probably be the methods using RGB-D cameras like ASUS Xtion [16; 17] and Microsoft Kinect [18; 19] fitted with infrared sensor to capture depth used for 3D reconstruction. Some algorithms used alternative methods of acquiring 3D data like the popular LiDAR, the laser scanner giving 360-degree view used in Google's self-driving car and Stanford's Stanley and even ultrasound sensors [20] have been used for this purpose. For a 3D reconstruction task, 3D data is really helpful for the tasks like the localization of the objects which can be performed directly in 3D using the depth information as in some of the large RGB-D datasets like ScanNet [7]. Although, these sensors have their advantages that they also provide depth in addition to the color information (RGB), they are not very cost-effective and also have a short range as compared to a regular camera which provides only RGB data in the form of images. Another disadvantage with the 3D sensors is that the methods that use them might not be able to perform in real-time since they require excessive use of GPU to process the 3D data and thus, the cost of implementation might grow exponentially with more and more 3D data captured while scanning larger scene for the reconstruction. On the other hand, the RGB cameras which are easily accessible and compact in size compared to the 3D sensors provide a very cost effective solution to perform the same task. The cameras capture images that are 2D projections of the 3D world loosing depth information in the projection. However, the ease of access of the regular cameras, their size and the relative cost make them really popular in the computer vision community who again finds the motivation in the human visual system which recovers and understands the 3D structure while only capturing the observations of the 3D world in 2D.

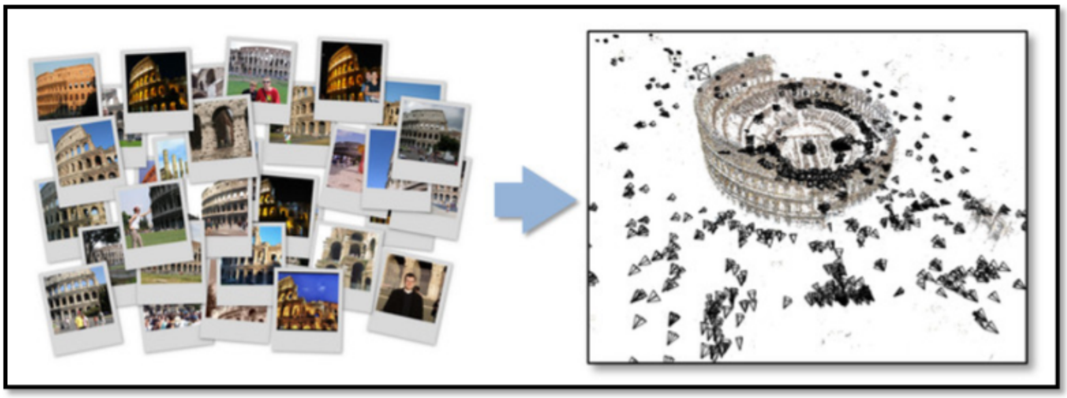


Figure 1.7: An example of Structure from Motion methods. Bundler [1] reconstructing the scene from unordered photos obtained from internet.

1.2.1 3D scene reconstruction

Over the years, there have been many developments in the field of extracting 3D information and reconstructing the scene using only 2D images obtained from the camera. Among these techniques, one of the very successful and popular methods is Structure-from-Motion (SfM) which uses multiple images captured from different view points to recover the pose of the camera for each view and 3D reconstruction of the scene in the form of sparse or dense point cloud. Some of the popular examples of the SfM systems are COLMAP [21], visualSfM [22] and Bundler [1] (Figure 1.7) etc. There are several steps in the standard SfM process which are: 1) extracting various interesting features from the images like corner points, edges or regions and match these points of interest in pairs of images. 2) The next step is to verify the pairs of images with common points of interests to guarantee the corresponding points found in the images also match the 3D geometry of the scene. This step serves as an initial reconstruction step and once, the images are geometrically verified the iterative part of the pipeline is initiated that takes in new images, remove outliers using triangulation and refine the reconstruction through optimization techniques like bundle adjustment [21].

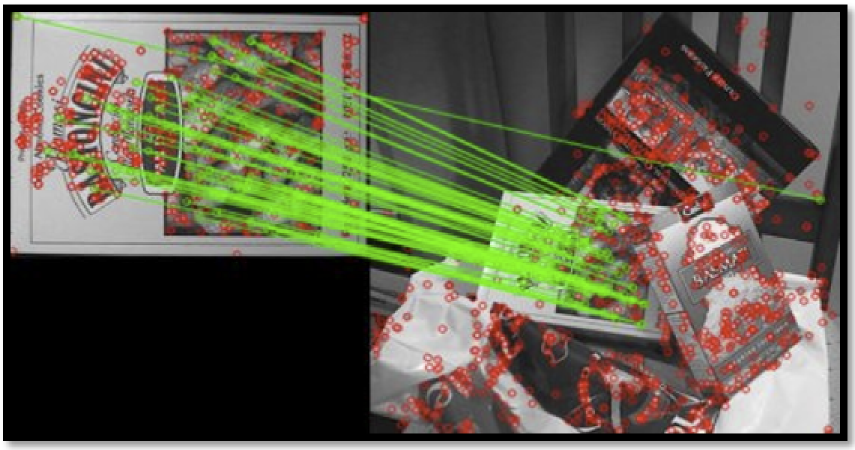


Figure 1.8: Feature matching using SIFT features extracted from a set of images shown as red circles in this image. The feature points extracted using SIFT are invariant to changes in scale, orientation and illumination.

1.2.2 Standard Structure-from-Motion

The building blocks of an SfM pipeline are further described in this section.

Points of interest or Features The main input to the SfM pipeline are the keypoints or point of interests or image features extracted from individual images to be later used for finding correspondences. Various solutions have been used in the literature for the feature extraction, most popular of them would probably be the scale invariant feature transform (SIFT) algorithm [23]. As the name is indicative, SIFT provides sufficient common feature points for correspondence which are invariant to changes in scale, orientation and illumination (see Figure 1.8). The presence of texture which is visually distinct and the resolution of images affect the number of extracted features and also, the accuracy of the correspondence.

Matching Features SfM method then tries to find correspondences between the set of point features obtained from different images in order to establish a relationship between the different region in the images. Based on the appearance

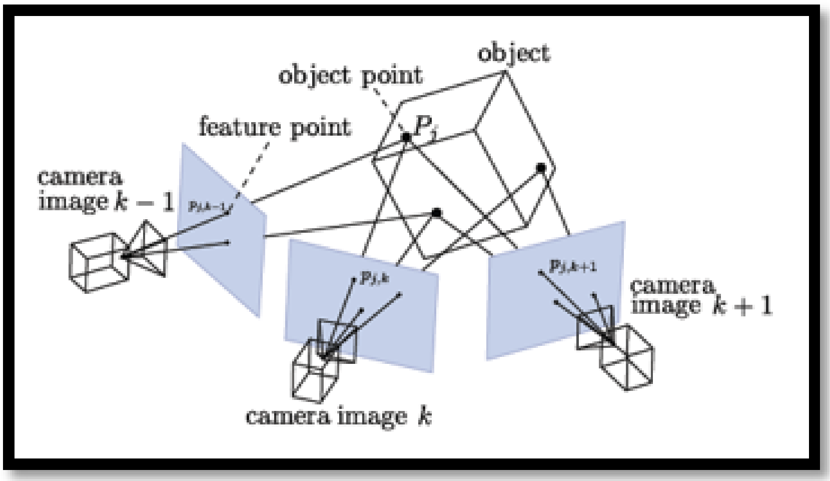


Figure 1.9: Feature points are matched in multiple views captured from, for example, the surface of an object in the 3D world. SfM methods such as shown here from openMVG [2] use epipolar geometry and triangulation to estimate the camera pose and the 3D coordinates of the points matched to generate a point cloud.

description, if feature correspondences are established between the features extracted from one region of an image to the ones extracted from a certain region of another image which is important as it establishes a common part of the scene between the two images. As shown in Figure 1.9, the feature points extracted from the projection of an object in multiple images is to be matched to ensure they belong to the same object or the same region of the scene. the set of images being observed could be a series of images by a single moving camera like in a video or it could be a set of different camera capturing the same scene from different view points.

Verification of matched points in image pairs Matching features and finding point correspondences between a pair of images is not good enough since it is also important to make sure that the matched point features also correspond with the 3D geometry of the scene. Since, many of these point correspondences might also be outliers, methods like RANSAC [24; 25] are used to remove them. Hartley and Zisserman [25] described two ways for geometric verification depending upon the spatial configuration of images captured. In case of a planar scene, the geometric transformation between two images can be given by homography and in the case of a non-planar scenes, the camera movement can be estimated using epipolar geometry with the essential matrix in case the camera's intrinsic parameters are known otherwise with the fundamental matrix. This verification leads to generating an image graph structure where the images constitute the nodes while the edges represent the geometrically verified pairs of images. The points matched in a verified pair of images provides the initial reconstruction in the form of a point cloud initializing the first two camera poses.

Image registration After an initial reconstruction is attained, new images are added to it by finding correspondences between the 2D feature points from the new images and the already known 3D points in the reconstruction obtained from the previous images. This 2D-3D correspondence to estimate the camera pose relative to a reference world coordinate system for a new image is known as image registration.

Triangulation The image registration step identifies the new images that contain feature points corresponding with the 3D points in the point cloud reconstructed so far. The new points are added to the existing 3D point cloud by a process called triangulation. Again, using epipolar geometry, the triangulation process estimates the 3D coordinates of every individual corresponding feature points between two images by using the relative camera poses of the two images. With triangulation, new feature points observed in the new images registered that can be added to the reconstruction which might lead to a denser 3D point cloud. In an ideal case, this addition of the new points into the reconstructed 3D point cloud would be accurate, however, that's not the case in practice since

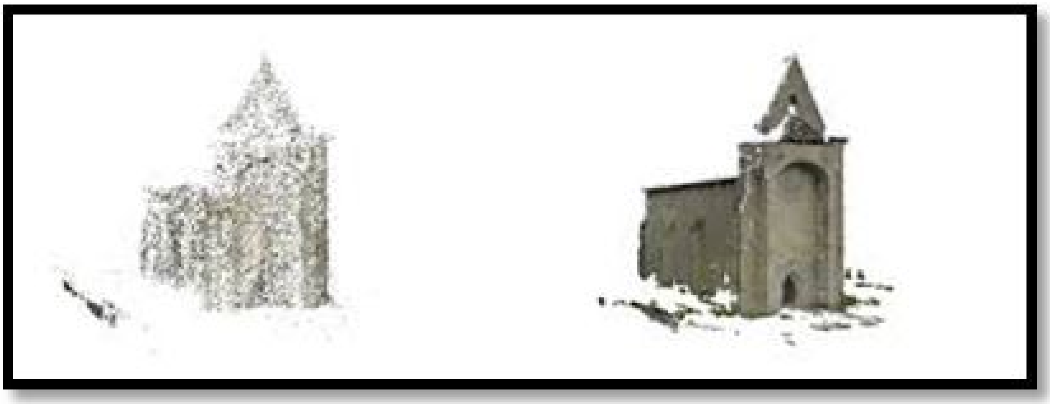


Figure 1.10: SfM methods provide 3D representation of the world in terms of sparse or dense point clouds [3].

there might be errors and inaccuracies accumulating from the previous steps till the triangulation phase.

Bundle adjustment To reduce the inaccuracies in the previous stages of the pipeline namely the estimation of camera poses and the new points added to the reconstruction by triangulation, the SfM method adopts optimization techniques like bundle adjustment [26] to minimize the accumulation of errors as the pipeline moves forward incrementally. Thus, an optimization technique such as bundle adjustment optimizes both the calibration parameters of the camera and the structure too by refining the reconstruction which provides an optimal 3D point cloud.

1.2.3 Object-based representation

Although, the SfM technique has been very popular and successful for the 3D reconstruction task, the representation of the 3D world has largely been limited to sparse or dense 3D point clouds (Figure 1.10). These point clouds may provide a great deal of geometrical information about a particular scene but this representation lacks any semantic information and thus, does not provide the crucial

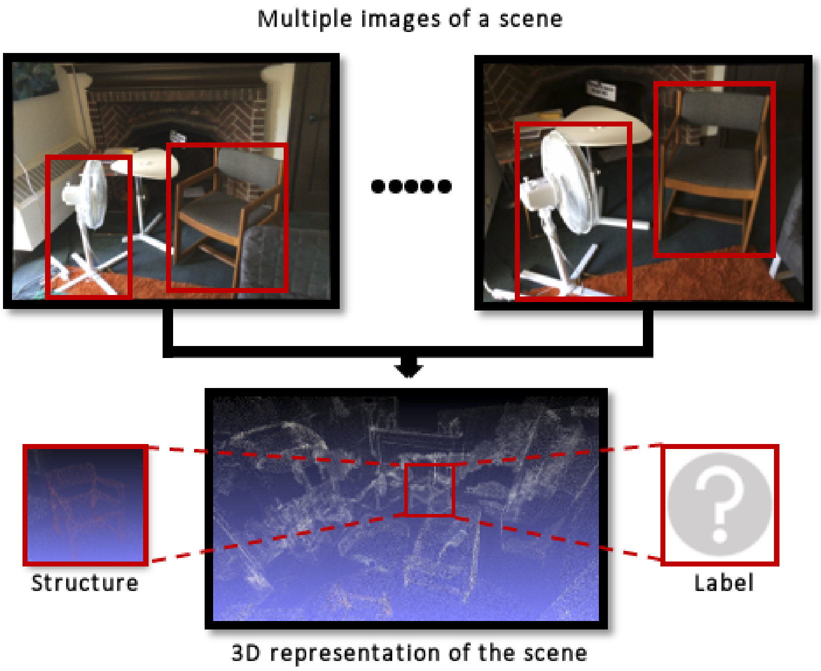


Figure 1.11: Sparse or dense point clouds generated from multiple images provide rich geometric information but they lack crucial semantic labels. The 3D representation shown here contains a semi-dense 3D point cloud. We can observe that the point cloud highlighted within the inset box belongs to the object class *chair*, however, a 3D point cloud such as this generated by an SfM method can only provide the structural details.

context in order to achieve a complete semantic scene understanding as shown in Figure 1.11.

For a better representation of the 3D world, we need to include the semantic information along with the geometric information in the models that we build of the 3D world. In the literature, a lot of methods have been studied for building an object-centric representation of the world that provides richer information about the scene in terms of the objects present in the scene. For example, [4] proposed a method to solve both the object recognition and online version of SfM known as simultaneous localization and mapping (SLAM) which resulted in a representation of the 3D world in terms of point cloud with segments recognized as 3D

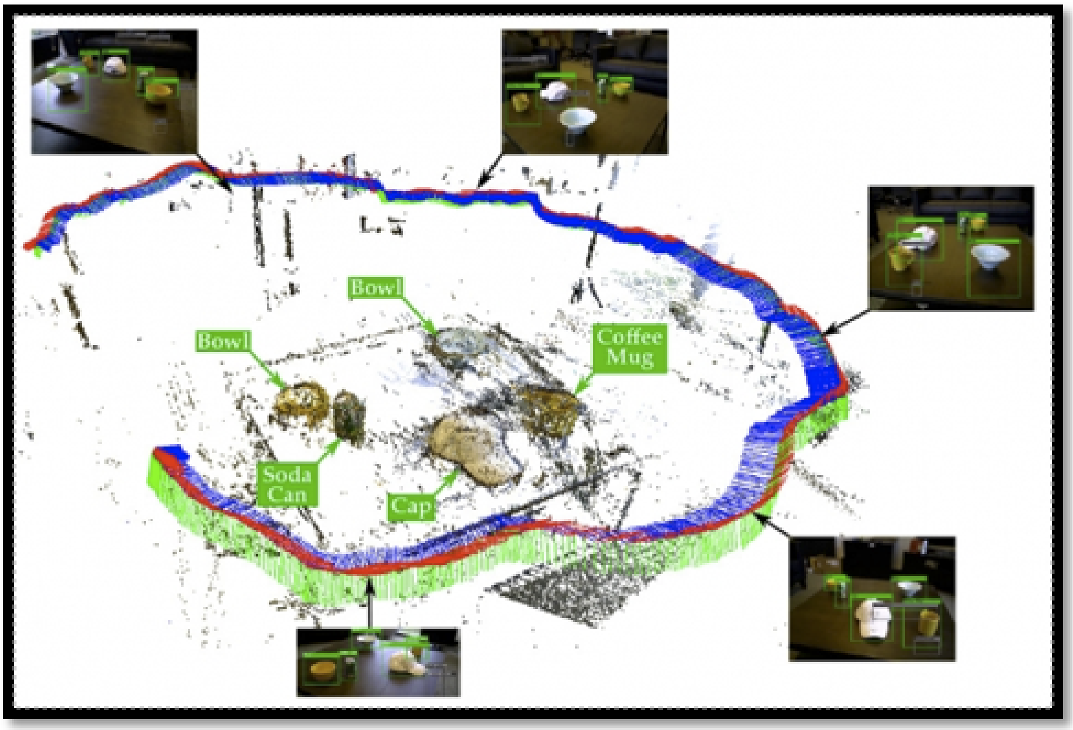


Figure 1.12: An example of an object-centric 3D reconstruction of a scene. [4]. In addition to the 3D point cloud of the whole scene, the representation also shows the point cloud segments belonging to the objects detected.

point clouds for the corresponding objects (Figure 1.12). Thus, this object-centric approach requires that the objects in the scene are detected and localized in 3D. Many times, the objects localized are shown by 3D bounding boxes depending upon the type of the data and semantic labels are assigned to them.

There are several methods [27; 28; 29; 30] in the literature that localize objects in 3D using the depth information obtained through 3D sensors. However, the aim of these techniques was not to accurately localize objects in 3D. A sliding window approach was used in [31] to scan over the whole space to generate 3D bounding boxes and then, classify each 3D bounding box using 3D CAD model renderings. However, this method was computationally expensive as sliding window over 3D space is very slow and demanding. Using the SUN-RGBD dataset, the methods like [32] propose to generate 3D bounding box proposals and classify these 3D bounding boxes using contextual features along with the oriented gradient descriptors and depth information. The other approach used is to parse the image into multiple segments and semantically label each segment pixel-wise. Lin et al. [28] proposed one such approach to use 2D segmentation in case of indoor scenes to generate bounding box proposals and used a conditional random field (CRF) to integrate the depth information and the data from different sources. There are other methods like [33; 34; 35; 36; 37] that parse images semantically into segments using RGB and depth information.

With the emergence of deep learning in computer vision, many algorithms have been developed over the years that use convolutional neural network for object localization in 3D. Some of the popular techniques are Fully Conventional Network [38] that provide pixel-level semantic segmentation and a deep learning based sliding window method that uses a 3D ConvNet [39] taking inspiration from 2D object detection techniques like the region proposal network (RPN) [40].

Again, working directly with the 3D data, 3D encoding of depth and 3D convolutions make these algorithms slow and computationally expensive and it also becomes hard to generalize them to all scene configurations if they use tools like 3D CAD model renderings, for example. As we have seen so far that the 3D reconstruction methods that need to localize objects in 3D use captured 3D data from specific sensors, they suffer from the challenges like missing 3D data, excessive computation and GPU usage. However, there have been a few methods recently developed that use the 2D data instead and gain a lot from the advancement of 2D object detection techniques with the deep learning networks. Faster R-CNN [40], Mask R-CNN [5] (Figure 1.13), YOLO [14] are some of the very popular 2D object detection methods that has become very robust, efficient and fast over the years. Apart from the fact that the 2D convolution process is much

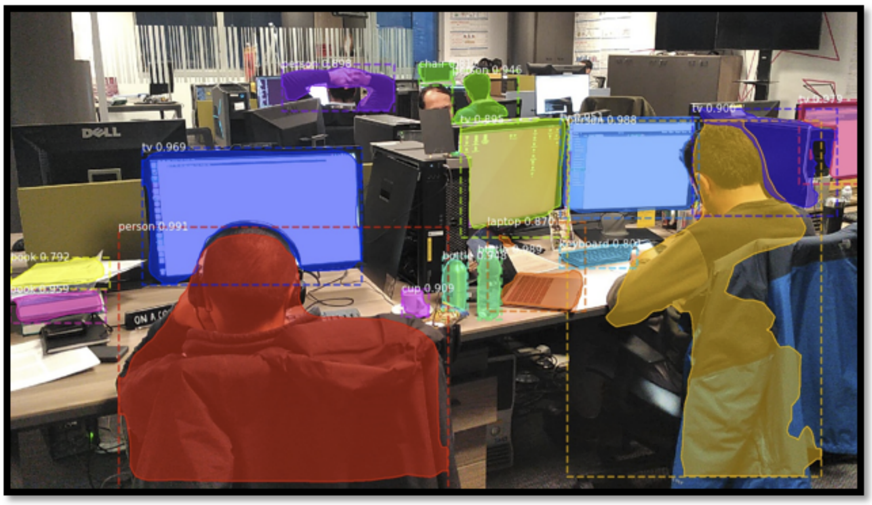


Figure 1.13: Objects detected in an image by Mask R-CNN [5] that also provides instance segmentation masks along with the bounding boxes.

faster than the 3D convolutions, the 2D data is much more consistent and reliable in comparison to the missing 3D voxels in a volumetric 3D representation of a scene. Since many recent 2D object detectors can very accurately detect thousands of object classes in the 2D images, it makes it easier to generalize over different scenarios consisting of various types of objects. Also, it is much faster and less computationally demanding to focus on specific regions bounded by 2D windows than the much more exhaustive search of the whole 3D area in the case of the sliding window approach.

One such technique is given by Lahoudv and Ghanem [41] that used 2D object detections to constraint the 3D projections in order to obtain tight 3D bounding boxes around the objects. The aim of the method is to place 3D bounding boxes over objects using the RGB-D data. The 2D bounding box when projected into 3D provided a much reduced search space in 3D by bounding the planes projected in 3D which are further constrained by the depth information to accurately detect objects in 3D.

However, there is another method that uses 2D object detections to extract 3D

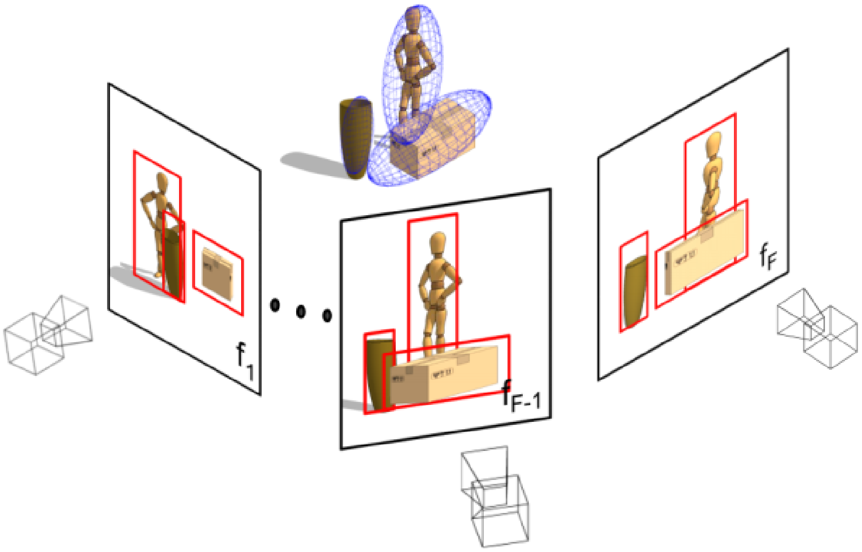


Figure 1.14: From the 2D ellipses fit to the object detection bounding boxes, the SfMO [6] method estimates the 3D ellipsoids in dual space that represent the position and occupancy of the objects in 3D providing a sparse reconstruction of the scene.

information without any need of an additional depth information. The method is aptly called Structure from Motion with objects (SfMO) [6] method which is also a closely related work described later in this thesis. Just like in the standard Structure-from-Motion (SfM) method where the 2D feature points are extracted, matched and are later used to estimate the camera pose and the 3D structure from multiple images, the SfMO method extends the approach to use 2D object detection instead of 2D points by developing a technique called localization from detection (LfD). Crocco et al. [6] and Rubino et al. [15] developed a method to recover objects' 3D position and occupancy from multiple view images of a scene using only 2D object detections. The problem was reformulated as the estimation of a quadric (ellipsoid) in 3D given a set of 2D ellipses fitted to the object detection bounding boxes in multiple views as shown in Figure 1.14. After having detections matched for all the images, 2D ellipses were fit to the bounding boxes and the localization of objects in 3D was instantiated as a quadric (a 3D ellipsoid) estimation from multiple 2D ellipses problem.

As we have seen, for complete semantic scene understanding, an object-centric 3D representation of the scene by localizing objects in 3D is necessary since the 3D point cloud alone doesn't provide any semantic information. This thesis will now discuss the methods we propose such as a probabilistic approach for 3D object localization with respect to the related work and why object instance re-identification is important.

1.3 Contributions

The following section summarizes the contributions of this thesis towards building such a semantic scene understanding system.

1.3.1 Re-identifying multiple instances of objects in indoor environments

The performance of a 3D scene reconstruction method such as the standard Structure-from-Motion using 2D feature points or an object-oriented SfM method such as SfMO [6], relies on extracting a good amount of geometric information from a set of different views. Just like the standard SfM techniques require to detect various 2D feature points in multiple images and find correspondences in multiple views, in a similar manner methods like Localization-from-Detection (LfD) [15] employed by methods like SfMO that work need to detect objects in 2D images and associate the 2D bounding boxes across multiple views. The LfD method takes advantage of a wide camera baseline where objects can be seen in different viewpoints. Different viewpoints provide richer information about the objects in those views which informs an accurate estimation of 3D quadrics. Thus, a very crucial step for the estimation of quadrics involves the the matching

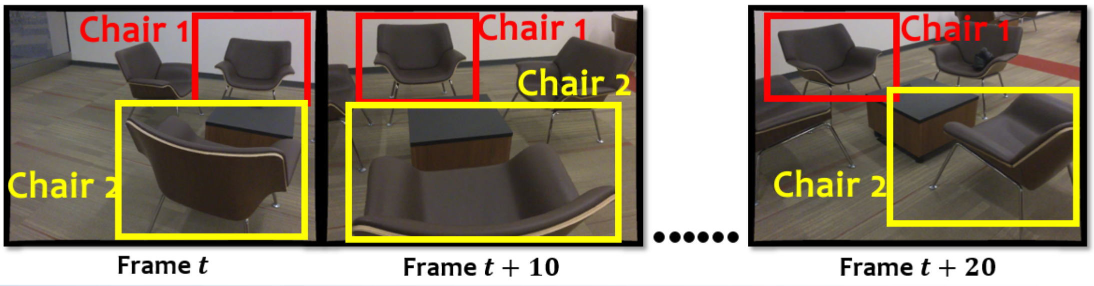


Figure 1.15: Similar looking objects in rigid, indoor scenes from ScanNet dataset [7]. Multiple instances of the same object class, chair, in this case, are hard to differentiate with each other. The goal of an object instance re-identification (re-OBJ) system is to be able to correctly identify different instances of the same object class in multiple views.

of 2D object detections in multiple frames. While working with SfMO method on images of an indoor environment provided by real-world datasets like ScanNet [7], we observed that a major reason behind the lower performance of SfMO on some of the scenes was poor data association of the 2D detections observed in multiple views.

We observed that given an indoor scene, where the environment is frequently cluttered with several near-identical objects, it is challenging to identify and track a particular instance of an object among a number of objects present in the scene (Figure 1.15). The problem is even more challenging when there is a wide baseline among multiple views (or temporally disjoint views).

Considering a static indoor video dataset where large displacement in the camera motion is unlikely and so the background of an instance cannot undergo a sudden drastic change. Therefore, we showed that in rigid scenarios, where the objects are stationary and only the camera is moving, it is not good enough to learn the appearance of an object (foreground) only but the background around the object is also important which can provide a lot of useful information regarding the surroundings of an instance of object which is unique to that instance at any given viewpoint. Described with more details in Chapter 3, we described a

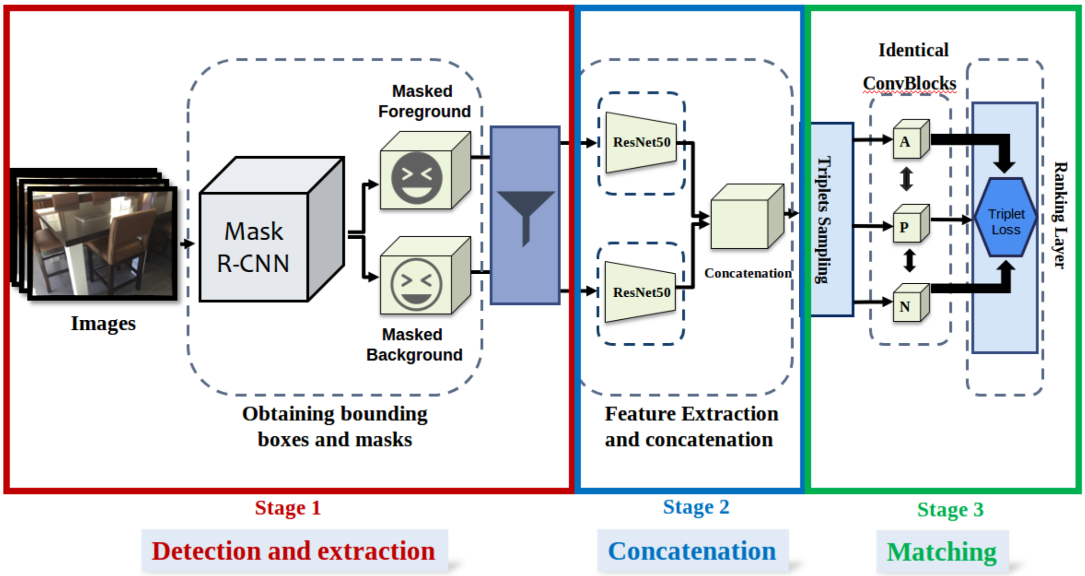


Figure 1.16: System description in three stages for the re-OBJ approach

novel method to jointly learn the foreground and background for object instance re-identification (re-OBJ).

To include the background information (Figure 1.16), the first step (Stage 1) in our approach is to use an off-the-shelf object detector like Mask-RCNN [5] and obtain foreground masks of the objects within the bounding boxes that are expanded in order to include a substantial background around the object within the bounding boxes. Encodings from the separated masked foregrounds and the masked backgrounds are extracted using ResNet50 [42], which are concatenated (Stage2) to obtain joint embeddings. These embeddings then are sampled into triplets$\{positive, negative, anchor\}$ and fed to a triplet-based network architecture (Stage 3) consisting of three identical ConvBlocks with the pairwise ranking model to learn image similarity for a triple-based ranking loss function.

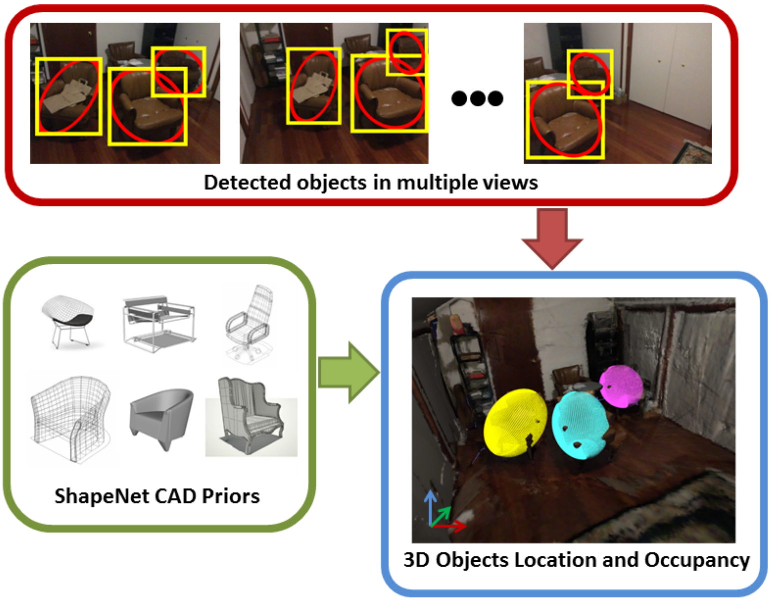

Detected objects in multiple views
ShapeNet CAD Priors
3D Objects Location and Occupancy

Figure 1.17: Given the object classes, the ShapeNet dataset [7] is used to create a realistic prior on the detected objects. Then, the Probabilistic Structure from Motion with Objects (PSfMO) method provides the metric localization, occupancy and pose of object as a set of quadrics in the 3D space.

1.3.2 Probabilistic framework to include object priors in SfMO

We have seen how the related work SfMO developed by Crocco et al. [6] and Rubino et al. [15] performs 3D object localization using the 2D detections. This thesis describes a way to extend the SfMO method in terms of extracting a more reliable estimate of the geometry in the direction of optical axis by including the object priors in addition to the object detections to estimate the object's position, occupancy and pose, called Probabilistic structure from motion with objects (PSfMO).

Described with more details in Chapter 4, PSfMO is a probabilistic framework to include the 3D objects priors to correct the ellipsoid axes lengths. As shown in Figure 1.17, given the ellipses in multiple views fitted inside the detection bounding boxes, firstly the SfMO method was applied to obtain the camera matrix and ellipsoid orientations were estimated. These values were used as an initialization for the PSfMO method. The matrix factorization used in SfMO could be framed inside the Probabilistic Principal Component Analysis (PPCA) [43] framework, thus enabling the inclusion of the object priors in the ellipsoid estimation. The object priors were given by the statistics on the dimensions of the objects collected by processing the CAD models from the ShapeNet dataset [44]. For each object category, the prior took the form of a 2D Gaussian that modelled the distribution of the ratio between the different object axes lengths.

1.3.3 An Augmented Reality (AR)-based embedded application

By combining the two proposed methods, PSfMO and re-OBJ, we show in Chapter 5 how an improved semantic 3D scene reconstruction can be utilized on an Augmented Reality (AR)-based platform such as a mobile phone device or a tablet. Such AR-based technologies allow users to render customizable virtual content over the real-world images they capture using the AR-enabled device. Discussed with more details in Chapter 5, we propose a pipeline to integrate both PSfMO for 3D object localization and re-OBJ for the correct association of the 2D bounding boxes in multiple views. We show our results on a sequence taken from a Tango-enabled mobile phone and show that the both the methods combined can provide an improved 3D scene reconstruction and thus, an enhanced AR experience.

1.4 Overview of the Thesis

This section provides an outline of the organization of remaining of the thesis.

Chapter 2 will provide an extensive insight into the previous research work and studies with respect to the study provided in this thesis on the topics of 3D object localization (Section 2.1). In this chapter, we will discuss how the attempt to 3D object localization would lead to the problem of multiple object association (Section 2.2) and how it can be handled by object instance re-identification (Section 2.3).

Chapter 3 will describe the method of object instance re-identification (re-OBJ) with details like the system design, triplet sampling, the loss function used etc. (Section 3.2) and will provide the experimental details including the training data (Section 3.3) and evaluation (Section 3.3.3).

Chapter 4 will provide details on the probabilistic structure from motion (PSfMO) method (Section 4.1) with experimental details (Section 4.2) with respect to the related work in the literature and an extensive evaluation with both synthetic (Section 4.2.1.2) and real-world data (Section 4.2.1.4).

Chapter 5 will discuss a real-world embedded application of the 3D quadric estimation for object's 3D location and occupancy using the PSfMO method described in Chapter 4 with an improved performance based on the re-OBJ method described in Chapter 3 using Tango application on a mobile device.

Chapter 6 will summarize the methods proposed in this thesis and their contributions of each of the components towards building a complete semantic scene understanding system. Also, based on each individual task, some potential future applications will also be discussed.

1.5 Publications

This section lists down the publications related to this thesis.

1.5.1 Conference

- Bansal, V., James, S. and Del Bue, A., 2019. re-OBJ: Jointly Learning the Foreground and Background for Object Instance Re-identification. In International Conference on Image Analysis and Processing (pp. 402-413). Springer, Cham. (related to Chapter 3)

- Gay, P., Bansal, V., Rubino, C., and Del Bue, A., 2017. Probabilistic structure from motion with objects (PSfMO). In Proceedings of the IEEE International Conference on Computer Vision (pp. 3075-3084). (related to Chapter 4)

1.5.2 Journal

- Bansal, V., James, S. and Del Bue, A., (To be published yet). Extension of re-OBJ with an improved architecture for a robust Object Instance Re-identification. (related to Chapter 3)

Chapter 2

Related Work

Summary

An autonomous system needs to build a 3D model of the scene it is observing. As we have understood in the previous chapter, to achieve this 3D representation which would also be meaningful we need to localize objects in 3D. Before we describe the method that we use for 3D localization using 2D data in the later chapters, we discuss in this chapter (Section 2.1) the extensive research work existing in the literature on the topic. As we discussed in Section 1.3.2, for a structure from motion pipeline that works with 2D object detections instead of 2D points it is crucial to associate the detections in multiple views. Section 2.2 will discuss the previous work in the literature review related to matching multiple objects in multiple views. We would also discuss the challenges of these related methods to perform the required task in the indoor scenarios especially re-identification of multiple instances of the object belonging to the same object category in multiple views in Section 2.3.

2.1 3D Object Localization

Initial work within the context of recovering 3D information from multiple-view images mostly involved the estimation of 3D position of the point correspondences extracted from 2D images[25]. Using point correspondences extracted from 2D images has inspired so many other research work like [45; 46; 47; 48] which use the standard structure from motion technique to estimate accurate 3D point clouds that are obtained from matched 2D point on the surface of realistic objects, even at a very large scale. However, the representation of the 3D world based on 3D point clouds is providing only a spatial information in terms of the 3D localization but is devoid of any semantic information as the context of the scene being reconstructed. On the other hand, the 3D localization of objects present in the scene can instead provide richer geometrical and much higher semantic information than a 3D point cloud which should, consequently, improve the performance of a classification or a recognition task in multiple views.

Localizing objects in 3D finds many practical applications like object manipulation using a robot[49; 50] and some classical computer vision tasks like Visual Question and Answering (VQA)[51; 52] and 3D-aware scene understanding [53; 54; 55; 56]. Some other previous work [57; 58; 59; 60; 61; 62] also emphasize how critical it is to utilize a higher semantic information that is provided by localizing objects in classical 3D reconstruction problems. This object-based semantic and geometric reasoning has been made possible now because of the accuracy and generalization of modern object detectors that can provide very accurate localization of objects belonging to various object classes in realistic scenarios[40; 63; 64; 65]. The approach that we have adopted and would discuss in a later chapter is a previous work SfMO [6], a Structure from Motion (SfM) method using 2D object detections obtained from a standard object detector instead of using 2D points as discussed in Section 1.1. The proposed work in this thesis, in comparison, uses objects' 3D shape priors in a probabilistic framework in order to obtain a reliable estimate of the geometry especially in the direction of optical axis where the original SfMO work lacks accuracy if the number of viewpoints with wide enough camera baseline is not available. There have been several examples in the literature providing probabilistic solution for SfM, mainly to improve the estimate of the 3D scene geometry. Forsyth et al. [66] recast the decomposition of the bi-linear components in factorization, camera matrices and 3D points coordinates, as a Bayesian inference problem. The motivation is to encode in the prior the metric constraints involved in the problem, thus providing better results in the presence of degenerate configurations of points. In face modelling problems, the work of Solem and Kahl [67] used a learned shape model to aid the 3D inference over regions for which there is no 2D information available. Del Bue et al. [68] used the information of the rigidity of some points to obtain reliable estimations of the 3D object structure with deforming objects. Information derived from object detections has already been used in SfM. The work described by Bao and Savarese [69] takes advantage of both semantic and geometrical properties associated with objects in the perspective case.

Another factorization problem that highly relies on priors is non-rigid SfM. This is due to the presence of objects 3D deformations that make the problem severely ill-posed. Torresani et al. [70] used Gaussian priors in a Probabilistic Principal Components Analysis (PPCA) framework together with a linear dynamic model over the deformation parameters. This framework is close to our method, however, our object representation enables us to build a better prior which is representative of a particular scene instead of a generic one. Similarly, [71] imposed a prior over temporal variations of the camera parameters combined with constraints over the proximity of projected 2D points and reconstructed 3D points. Again related to 3D points estimation, [72] defined a shape prior in a factorization based approach to help 3D reconstruction in case of degenerate motions. Akhter et al. [73] showed that a prior parametrization of the 3D trajectory motion can provide more efficient results. The work of Gotardo and Martinez [74] proposed a similar principle using DCT bases to represent the camera motion in order to regularise intrinsic and extrinsic parameters. Finally, [75] used a novel Procrustean Normal distribution to minimise geometrical deformations under an optimality criterion.

All these approaches deal with 2D point trajectories or matches in multi-view, only few work directly localise objects in a factorization framework. Previous methods attempted the joint reconstruction of different geometrical entities such as lines [76; 77], curves [78; 79; 80; 81] and conics [82; 83; 84]. However, even if these methods were able to obtain an inference of the 3D structure, the goal of these methods was not an object-based representation of the 3D world. Recently, the work of Crocco et al. [6] proposed the SfM with Objects (SfMO). This method provides a solution to the 3D localization of objects in a factorization framework by using the output of detectors only as is described in Chapter 1, Figure 1.13. However, even if the method provides a closed-form solution, it can lead to unreliable estimates, especially for the object occupancy, if the detector output is not accurate enough or if very few views are available with limited camera baseline.

The proposed work in this thesis (Chapter 4) is a probabilistic extension of the SfMO method where we refine the estimation of 3D quadrics by using 3D object shape priors which improves the accuracy in the direction of the optical axis and provides a better estimation of the objects' occupancy in 3D. But, in order for this method to work, the object detections need to be matched across the multiple views. In a real-world scenario, the environment might have multiple objects with similar appearance or even multiple instances of the same object that are all to be matched across the multiple views. In a similar way that the point correspondences are utilized to estimate the 3D position of the points in the world coordinates in the standard Structure from Motion (SfM) method, 2D object detections are used for the 3D object localization in SfMO [6] and its extension that we proposed in PSfMO. Since we need to associate the correct bounding boxes for each object, the multiple object association is necessary across the multiple views. When this multiple object association is performed for the already seen object instances across different views, especially when the camera revisits the same region of the scene after a long time, we define the problem as object instance re-identification. The next section will first discuss the related work in the literature regarding the multiple object association and how the challenges face by them eventually lead us to the task of object instance re-identification (Chapter 3).

2.2 Multiple Object Association

Conventional methods use two major approaches to build a re-identification system: appearance-based and motion-based. Most methods use an appearance-based approach because motion prediction based systems try to localize each object instance based on a motion model, however, due to the possibility of huge unpredictable trajectories across the camera views, these methods tend to fail when the same object instance reappear after a long time.

2.2.1 Appearance-based

Many image similarity models [85; 86; 87] simply extract features like Gabor filters, SIFT [88], HOG [89] features to learn similarity between images. However, the representation of the hand-crafted features limited the performance of these methods since the accuracy of the methods detecting these features in the images

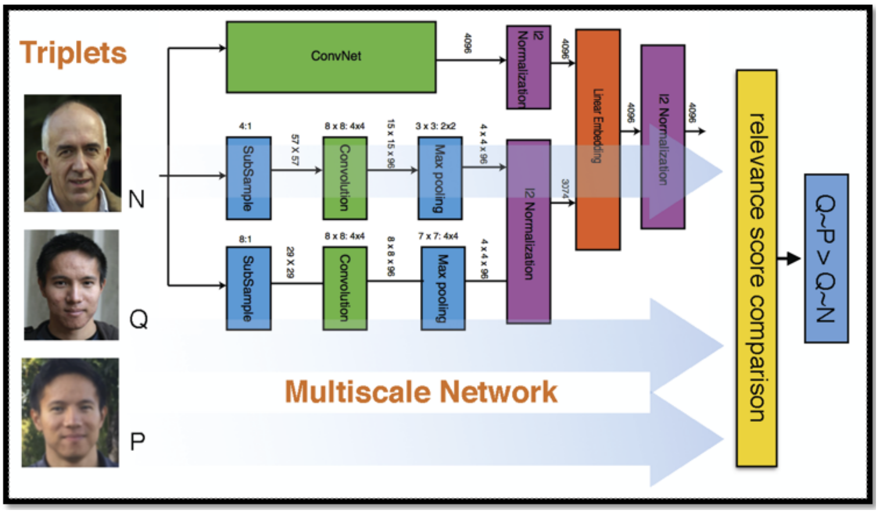


Figure 2.1: The architecture of a triplet-based model used by Wang et al. [8] to learn a fine-grained image similarity function for both, inter-class and intra-class variations in images.

vary with the datasets. Some other previous studies work on finding similarity in images [90; 91] where they are considered based on the category they fall under. On the other hand, deep learning methods like the Convolutional Neural Network (CNN) need not to be provided such features instead they can learn them from the images. Some deep learning-based models popular in the image classification tasks [92] have shown great success in learning features from the images. For example, Nguyen et al. [93] showed that the learned deep features perform better than the hand-crafted ones in a face recognition task. But, even these deep models cannot directly fit similar image ranking especially the fine-grained distinction between similar images. Thus, in order to learn a fine-grained image similarity function, a deep ranking model (Figure 2.1) was proposed by Wang et al. [8]. Pairwise ranking model is a widely used learning-to-rank formulation especially in the image retrieval task. In the image retrieval methods like [94; 95], the goal of learning-to-rank is to learn a ranking function that extracts the most relevant images in the top-k results when probed by a query image based on userpreferences. Learning-to-rank has also been used to learn image ranking models in [86; 96; 97]. This learning-to-rank model for finding similar images becomes the foundation for our work for object instance re-identification explained in Chapter 3.

Other deep learning-based methods find similarity in the images based on the category they fall under. Taylor et al. [87] developed their method that finds semantic similarity between a pair of images to find if they both belong to the same category or not. Friedman and Russell [98] explored the relationship between visual and semantic similarities where they found that it is possible that the two images that are semantically similar (i.e. belonging to the same category) might differ visually.

Thus, applications that build upon image similarity like re-identification, image retrieval, search-by-example etc. require learning a fine-grained image similarity function that can also distinguish the differences between visually different images of the same category. Thus, the appearance-based approaches discussed so far might be good at distinguishing inter-class or even intra-class variations in the images, but for our task where we need to associate two similar looking object instances of the same semantic class in multiple views, these methods trained only on the foreground appearance tend to fail. In the scenario of an indoor scene that we consider in our work, the multiple instances of the same object category are both visually and semantically similar. Hence, compared to the previous work, we will focus on the instance's relationship to its background to jointly learn a foreground and background discriminative feature as described in Chapter 3.

2.2.2 Motion-based

A number of methods in the literature address the problem of matching multiple objects by detecting motion across the multiple views. For the static scenes or in other words, the environments where the camera is fixed, there has been a trend to utilize methods based on temporal averaging of an image sequence [99; 100] and video object segmentation based methods [101] where the aim is to segment the objects' foreground achieved by analyzing the motion [102; 103] and clustering trajectory [104; 105; 106]. In [102], Faktor and Irani proposed a method to estimate the motion salient regions by identifying dominant motion present in the scene. The saliency scores are obtained from estimating the motion difference against the detected dominant motion. On the other hand, Papazoglou and Ferrari [103] identify motion salient regions using optical flow by detecting the motion boundaries. Some recent works like [107; 108; 109] utilize deep learning based methods in an unsupervised manner for finding motion patterns in the sequence of images.

If the target for the motion detection is an object within a video then Multiple Object Tracking (MOT) becomes the most popular application for a motion-based object association method. The Multiple Object Tracking (MOT) task is to track a target object and predict its position in the successive views in a static or dynamic environment captured in a video sequence. One very popular technique to predict the motion of a target is Kalman filter [110] which uses the change in the state of the target from the previous point in time to the current one for the prediction of its future state. Also, a very popular technique to describe motion across different camera views within a video is optical flow developed by Lucas and Kanade [111].

Another popular framework for MOT is tracking-by-detection which takes advantage of the tremendous development in the field of object detection in the past few years. Previous methods like Bochinski et al. [9] detect objects in the images using an object detector and then, associate the bounding boxes in multiple views to estimate the trajectory of a targeted object as shown in Figure 2.2.

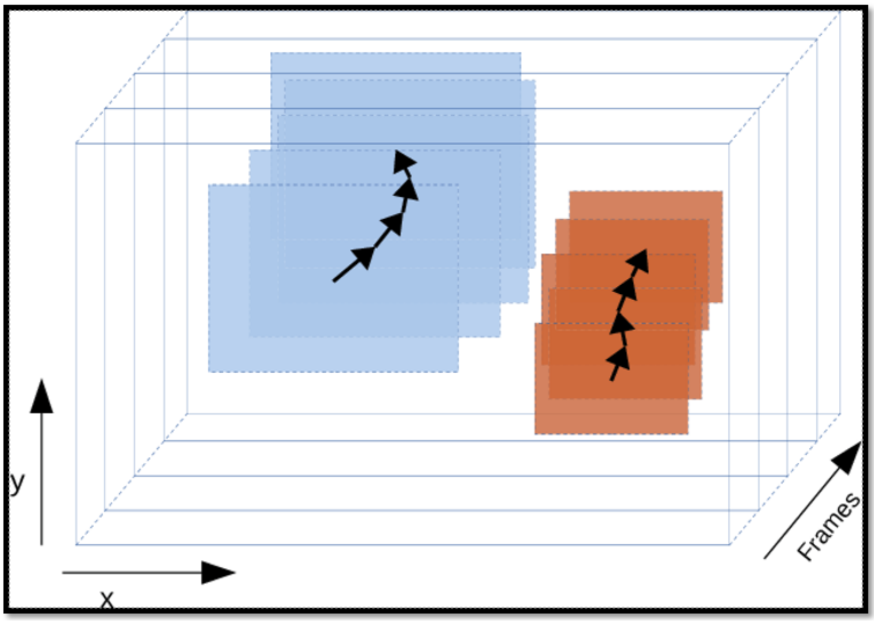


Figure 2.2: The tracking-by-detection methods like proposed by Bochinski et al. [9] estimate the trajectory of the bounding boxes in multiple views. For the estimation of trajectory, these bounding boxes need to be associated in all the views.

However, methods that estimate motion using Kalman filter, optical flow or even tracking-by-detection might fail in the case where there is a sudden change in the camera motion in the video. The tracking-by-detection relies heavily on the performance of the object detectors which might not detect many objects in all of the images due to challenges like occlusion, motion blurr or low resolution of the images.

Many methods used deep learning to solve these challenges. To handle challenges like the drift accumulated by occlusion etc., Chu et al. [112] proposed a spatial-temporal attention framework in a single object tracker settings. Methods like deepSort [113] combined an appearance model based on the deep features extracted from the bounding box of a detected object with the motion information to build a deep association metric for matching the objects (pedestrians, in this case). For the real-world scenarios that suffer from challenges like occlusion, camera jitters and unpredictable trajectory changes, the methods based on motion segmentation or motion prediction tend to fail in a robust object matching task. For our task, where we need to differentiate and re-identify the already seen object instances in multiple views, even methods like deepSort [113] that employ appearance-based model in combination with a motion model but there appearance-based model suffer the same challenges as are described in Section 2.2.1. In almost all of the object tracking methods, the switching IDs is the most common problem where the tracking algorithm switches the target's tracking ID from one object to another mostly because of the unpredictable camera motion leading to poor data association from frame to frame.

Thus, the multiple object tracking methods try to locate and track a feature, a segment or a bounding box, depending upon the principle used, in different views by utilizing the frame-to-frame temporal information. This is in contrast with our problem of re-identifying multiple instances of an object in different views of a rigid scene. The current motion-based object tracking methods in the literature would fail because of the challenges explained so far such as unpredictable camera trajectory particularly in the scenario where the camera revisits the same area of the scene after a long time while our proposed method would re-identify the same instance already seen before given a similar point of view.

2.3 Object Instance Re-identification

There is a vast literature for object re-identification that is mostly focused on person re-identification where the goal is to assign a correct ID of an instance of a specific class (i.e. a pedestrian) across multiple-views obtained from cameras with possibly non-overlapping views. In general, these methods try to learn discriminative features based on person's face [114], clothing [115] or symmetry-driven local features [116] to re-ID people. In contrast, the problem of associating a unique ID to instances of objects is often solved as the association of multiple unknown objects between views [117]. This problem is closely related to person re-ID and is often evaluated in the pedestrian (person) scenario with early work on PET2009 [118]. There are many other re-identification methods that use appearance-based object association as discussed in Section 2.2.1. To re-identify objects in the images such methods heavily rely on finding a similar set of images for a given image of the target object using visual search to retrieve similar images to the given query image. Some work in the literature like [116; 119] exploit the knowledge that the same individual is been detected in consecutive frames and then learning an appearance-based transfer function for a robust re-identification system. Additionally, in [116], Farenzena et al. extract features from three different complementary modalities: the chromatic content, spatial arrangement of colors and local motifs derived from different parts of the human body to accumulate local features. FaceNet [114] showed that the recognition of a human face could be improved using a triplet loss function which is more suitable for the verification, recognition and clustering than the verification loss [120]. The difference is that the verification loss minimizes the $L2$-distance between objects of the same identity and enforces a margin between the distance of objects of different identities whereas the triplet loss also encourages a relative distance constraint to discriminate between dissimilar identities.

However, the object instance re-identification we discuss in this thesis is different than the pedestrian/person re-identification. The specific task of re-identifying multiple near-identical objects in a rigid scene presents a different challenge, we refer to as re-OBJ, a specific case of re-ID. A closely related work, RIO [121], introduced a method for object instance re-localization in 3D. They

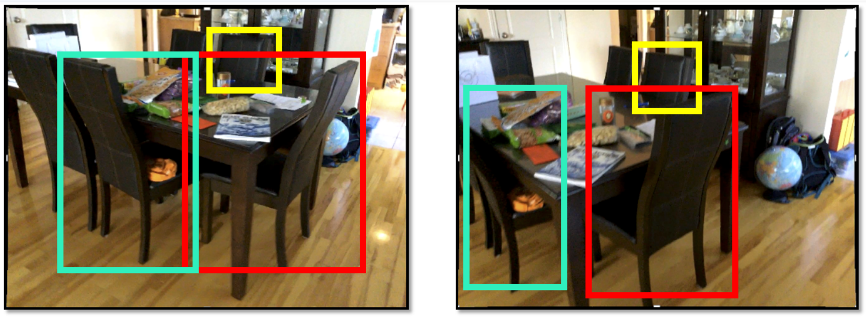


Figure 2.3: The problem in a real-world scenario is to identify different instances of an object in multiple views. An important observation is that in a video dataset, the background does not change a lot with time as can be seen with these two adjacent views of a scene.

use a fully-convolutional 3D correspondence network to find matching features related to multiple objects in changing 3D scans in order to estimate their corresponding 6DoF poses in another scan of the same indoor environment differed by time.

We consider a static indoor video dataset where large displacement in the camera motion is unlikely and so the background of an instance cannot undergo a sudden drastic change. Therefore, we propose to jointly learn the foreground and the background to build a robust object re-identification system at the instance level. We propose not only to learn the appearance of an object's foreground but also the background that can provide a lot of useful information regarding the surroundings of an instance which is unique to that instance at any given viewpoint. Consider a scene of a dining room with multiple chairs present around a table as shown in Figure 2.3. To re-identify a particular instance across multiple images, it is important to be able to distinguish it from other instances of the same object class. Intuitively, if we can observe and encode the surroundings of that instance within a stream of images, we can be confident to an extent that the target object instance has been seen before and it is different from other instances of the same class because the environment around it is unique at any given point of time even when other instances have similar appearance. Our work is inspired from the deep ranking model proposed by Wang et al. [8] with an efficient triplet sampling algorithm where we sample different object instances into triplets as described in Section 3.2.1 of Chapter 4.