Multiple view geometry is the subject where relations between coordinates of feature points in different views are studied. It is an important tool for understanding the image formation process for several cameras and for designing reconstruction algorithms. Multiple view geometry can be broken down into fundamental tools. The first to be discussed is projective geometry. Projective geometry is a vital tool when it comes to dealing with structure from motion problems in computer vision, especially when it comes to multiple view geometry. The reason for this is because that the image formation process can be regarded as a projective transformation from a three dimensional space to the two dimensional space. Furthermore, it plays a part in dealing with problems in camera calibration, critical motion sequences and examining critical configurations.

We will start the introduction of projective geometry by looking at a central perspective transformation

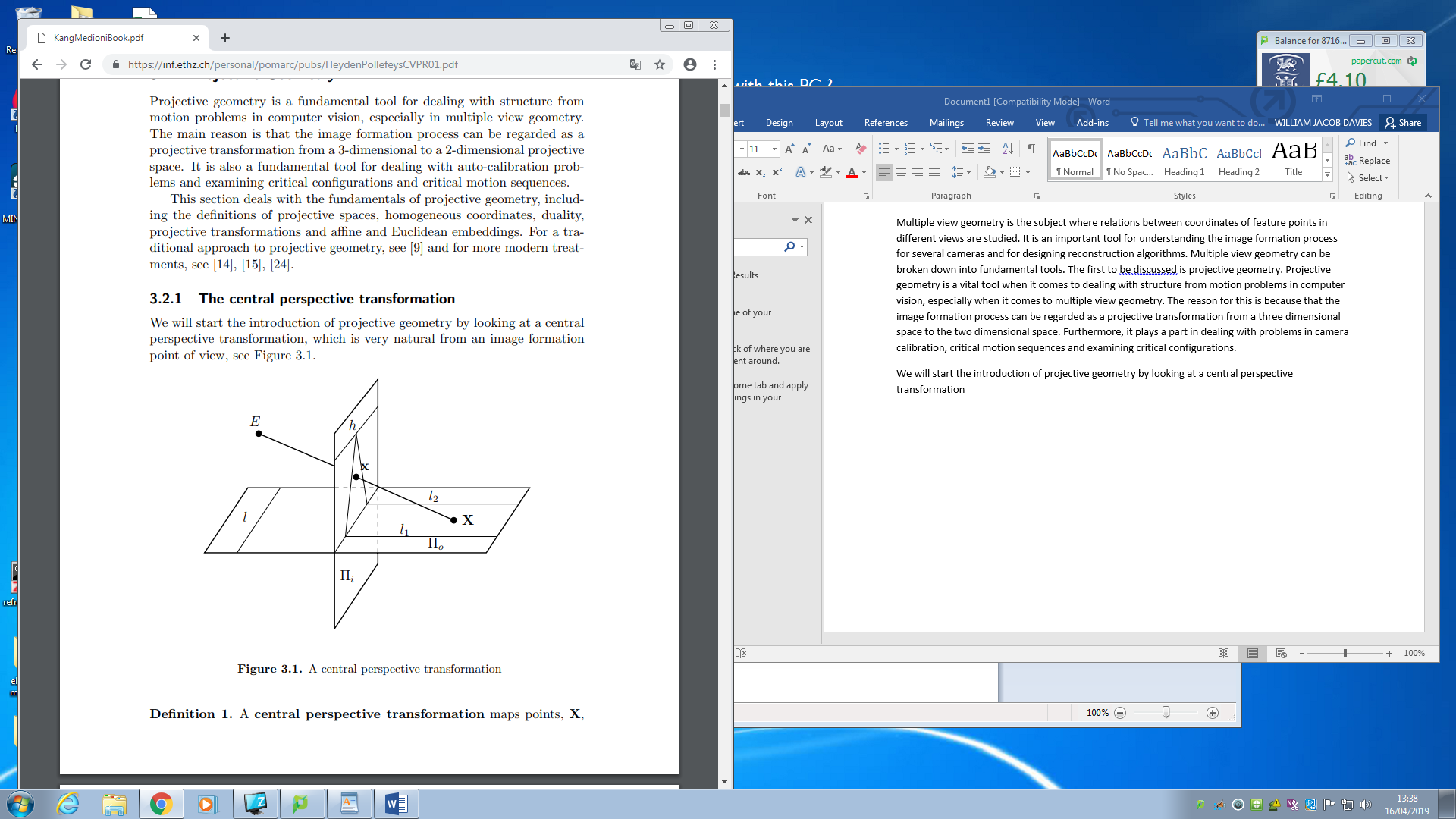


Figure 1 – A central perspective transformation

A central perspective transformation will map the points ,X, which are on the object plane IIo to points on the image plane Iii via and intersection through E. From the above figure 1, we can observe the following properties of the planar perspective. Firstly, All points on Πo maps to points on Πi except for points on l, where l is defined as the intersection of Πo with the plane incident with E and parallel with Πi. Secondly, all points on Πi are images of points on Πo except for points on h, called the horizon, where h is defined as the intersection of Πi with the plane incident with E and parallel with Πo. Thirdly, the lines of Πo are mapped to lines Πi. Fourthly, the images of parallel lines intersects in a point on the horizon as exampled by l1 and l2. Finally, in the where the point E moves infinitely faraway the planar perspective projection becomes a parallel projection.

Now we can look at this in regards to a standard Cartesian coordinate system. This will identify the planes Πi and Πo

SLAM

Why is slam used, it is central to a range of indoor and outdoor, air and watered applications for both manned and autonomous vehicles. Examples of this would be a vacuum cleaner (dyson), air surveillance, reef monitoring, exploration of mines and terrain mapping

Slam is used to compute the different positions of the robot in different points in time. In order for this to be achieved, localization and mapping must be solved simultaneously together.

Localization is estimating the robots point in time. The robots position is based on landmarks. Landmarks are something that the robot can recognise. Its position is based on where the robot is relative to the landmark. Mapping is constructing the environment in which it is moving through. As it, moves through it will map each feature (landmarks) location. The mapping of the features will not be spot on and this can be due to noisy sensor data. Mapping uses a Gaussian distribution in or der to calculate the probability of there that landmark is.

How to start solving the problem

We must first look at what we know and what we desire.

What we know – a series of commands given to the robot which will be defined as

U1:T = {u1,u2,u3….,ut}

Observations which can be defined as

Z1:T = {z1,z2,z3….,zt} : note the notation 1:T signifies different points in time.

Given these parameters the goal is to obtain a map of the environment

The path the robot has taken which can be defined as

X0:T = {x1,x2,x3….,xt}

Note the reason why x has one index more is that transitioning from three point’s only commands in that sequence would be executed. For example I have points a b c which is my path variables. One transition from a to b then b to c. These discussed variables are not without fault. Most of the time a probabilistic approach is taken in order to say the robot is in this region. These probabilities can be calculated by the following :

P(xo:t,m|z1:t,u1:t)