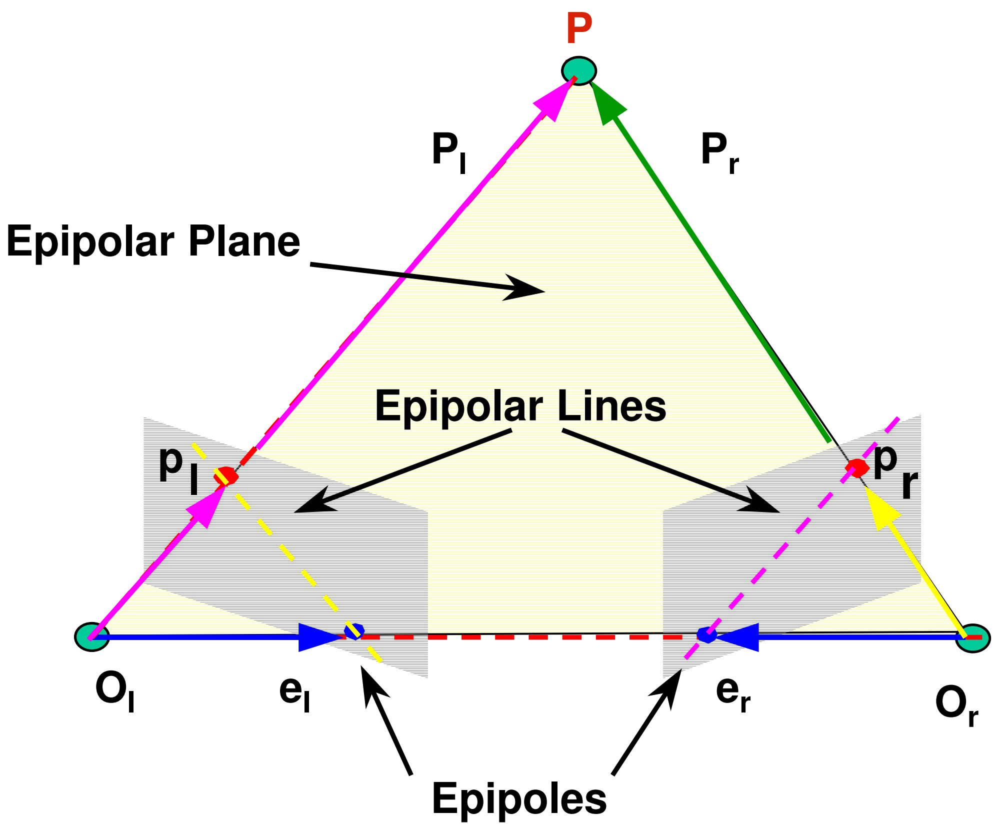
Homography

Dr. Gerhard Roth

Epipolar Geometry



$$\mathbf{P_r} = \mathbf{R}(\mathbf{P_l} - \mathbf{T})$$

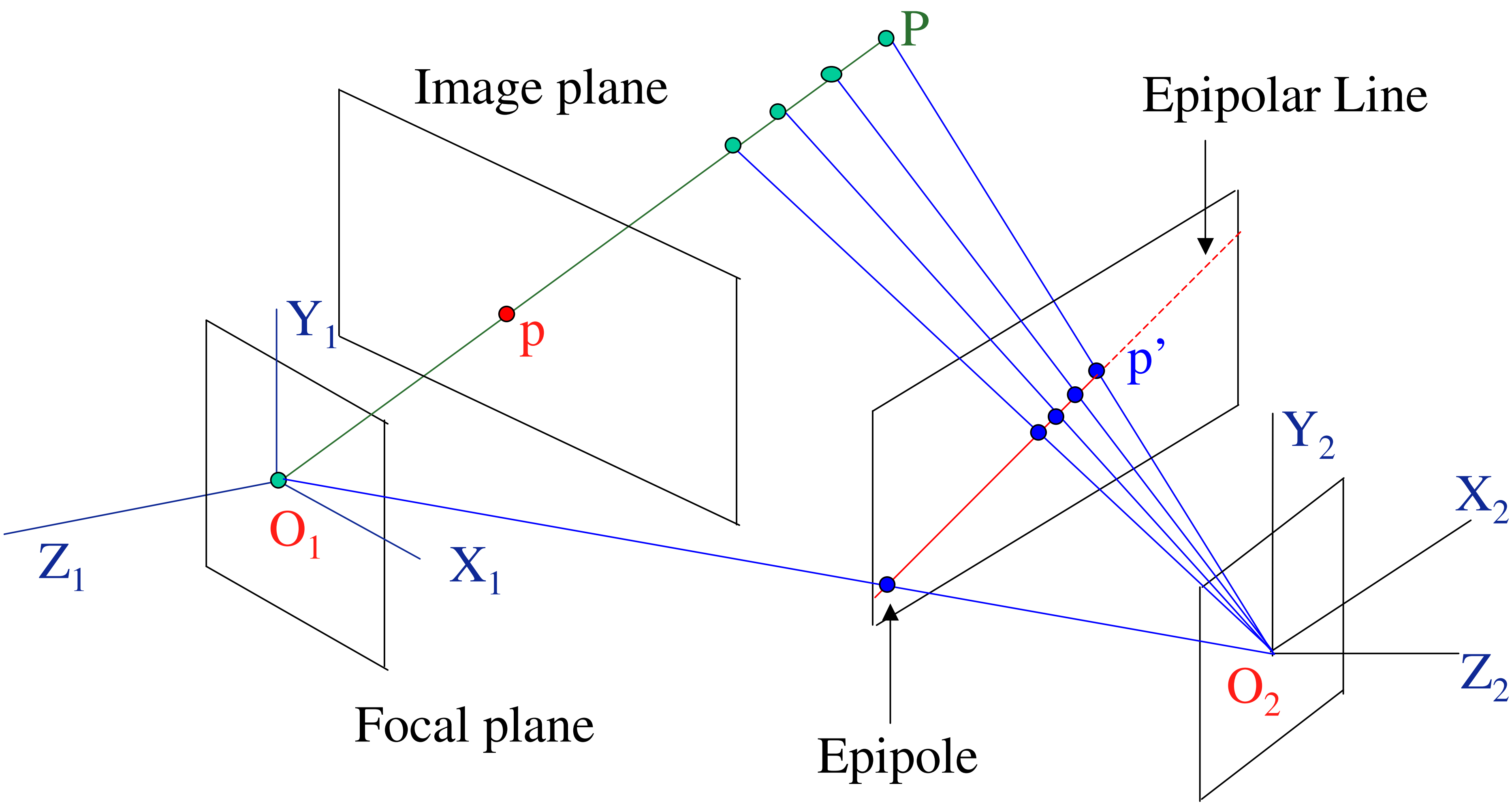
Homography

- Consider a point x = (u,v,1) in one image and x'=(u',v',1) in another image
- A homography is a 3 by 3 matrix M
- $$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}$$
- The homography relates the pixel co-ordinates in the two images if x' = M x
- When applied to every pixel the new image is a warped version of the original image

Homography conditions

- Two images are related by a homography if and only if
- Both images are viewing the same plane from a different angle (your assignment)
- Both images are taken from the same camera but from a different angle
  - Camera is rotated about its center of projection without any translation
- Note that the homography relationship is independent of the scene structure
  - It does not depend on what the cameras are looking at
  - Relationship holds regardless of what is seen in the images

What does Essential Matrix Mean?



Homography for Rotation

- If there is no translation $P_r = RP_l$
- Home position – projection equation

$$x = K[I|0]\begin{bmatrix} X \\ 1 \end{bmatrix} = KX$$

- Rotation by a matrix R – projection equation

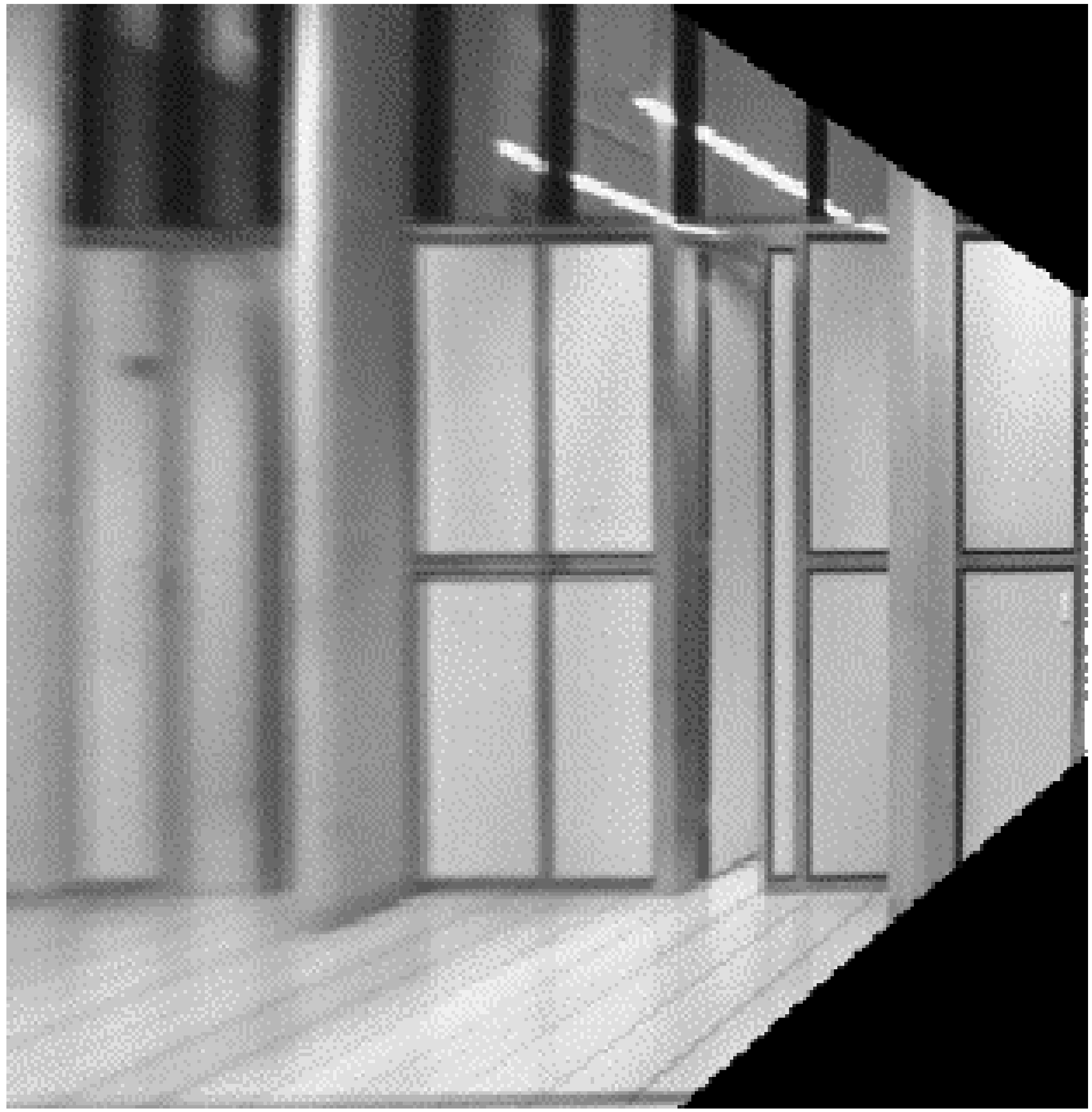
$$x' = K[R|0]\begin{bmatrix} X \\ 1 \end{bmatrix} = KRX$$

- So $x' = KRK^{-1}x$ where K is calibration matrix
- Where KRK^{-1} is a 3by3 matrix M called a homography

Original camera view

Downward rotation via homography

Sideways rotation via homography



Homgraphy versus epipolar geometry

- If two cameras are translated then epipolar geometry holds
  - It maps a point in one image to a line in the other image
  - Actual matching point depends on the depth of that point, that is its distance from the camera
- As the translations -> 0 the baseline b of the two cameras also goes to zero
  - Once it is zero epipolar geometry does not hold any longer
- Homgraphy holds when no translation
  - It maps a point in one image to a point in the other image
  - The mapping does not depend on the depth (distance) of point to the camera
  - This makes sense because baseline z zero means ordinary stereo triangulation equations fail

•Computing homography

•If we know rotation R and calibration K, then homography M can be computed directly $x' = KRK^{-1}x$

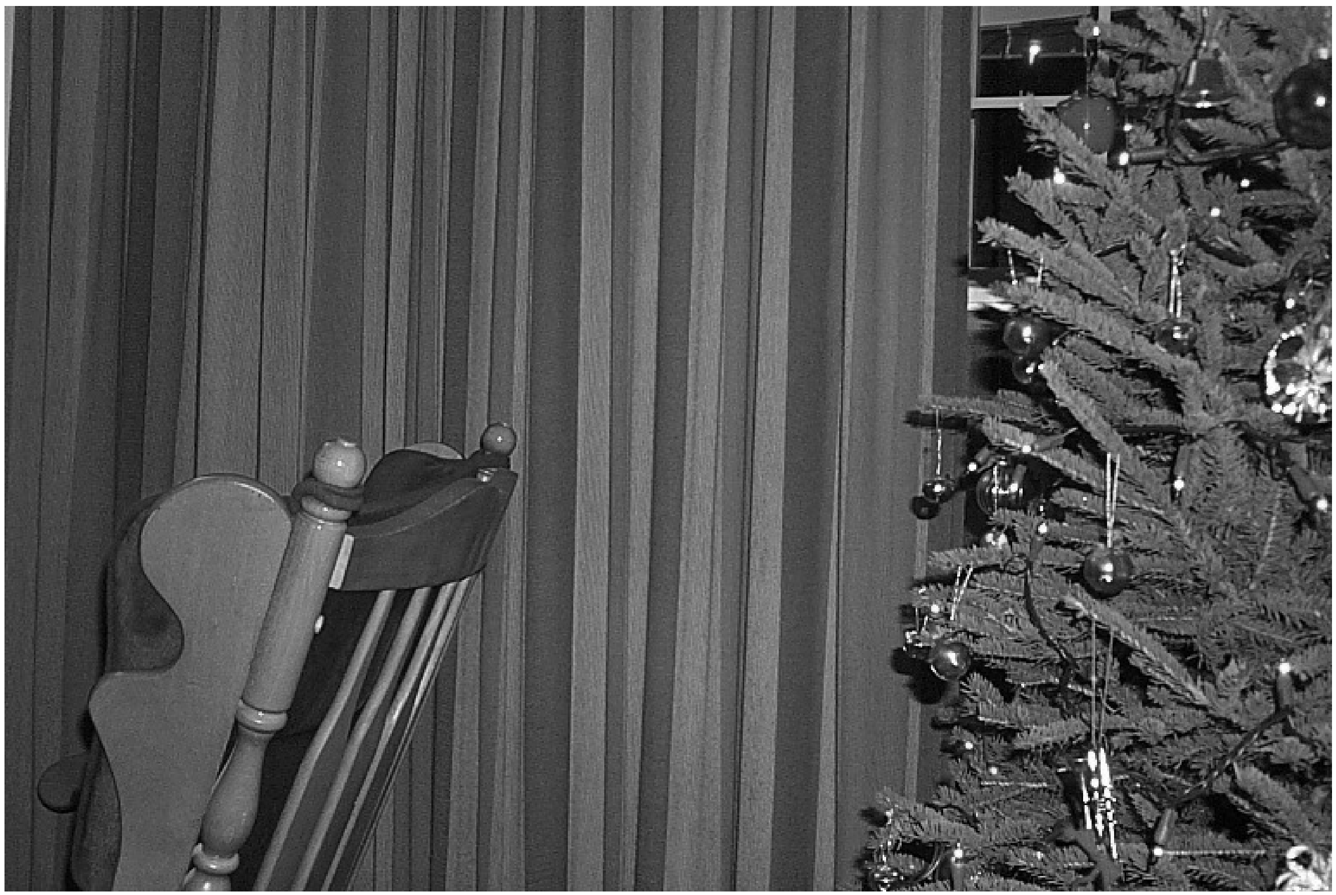
- Applying this homography to one image gives image that we would get if the camera was rotated by R
- Inverting M, to get M^{-1} is same as applying inverse rotation R^{-1}

•But if we have two rotated images but do not know the rotation then how can we compute the homography?

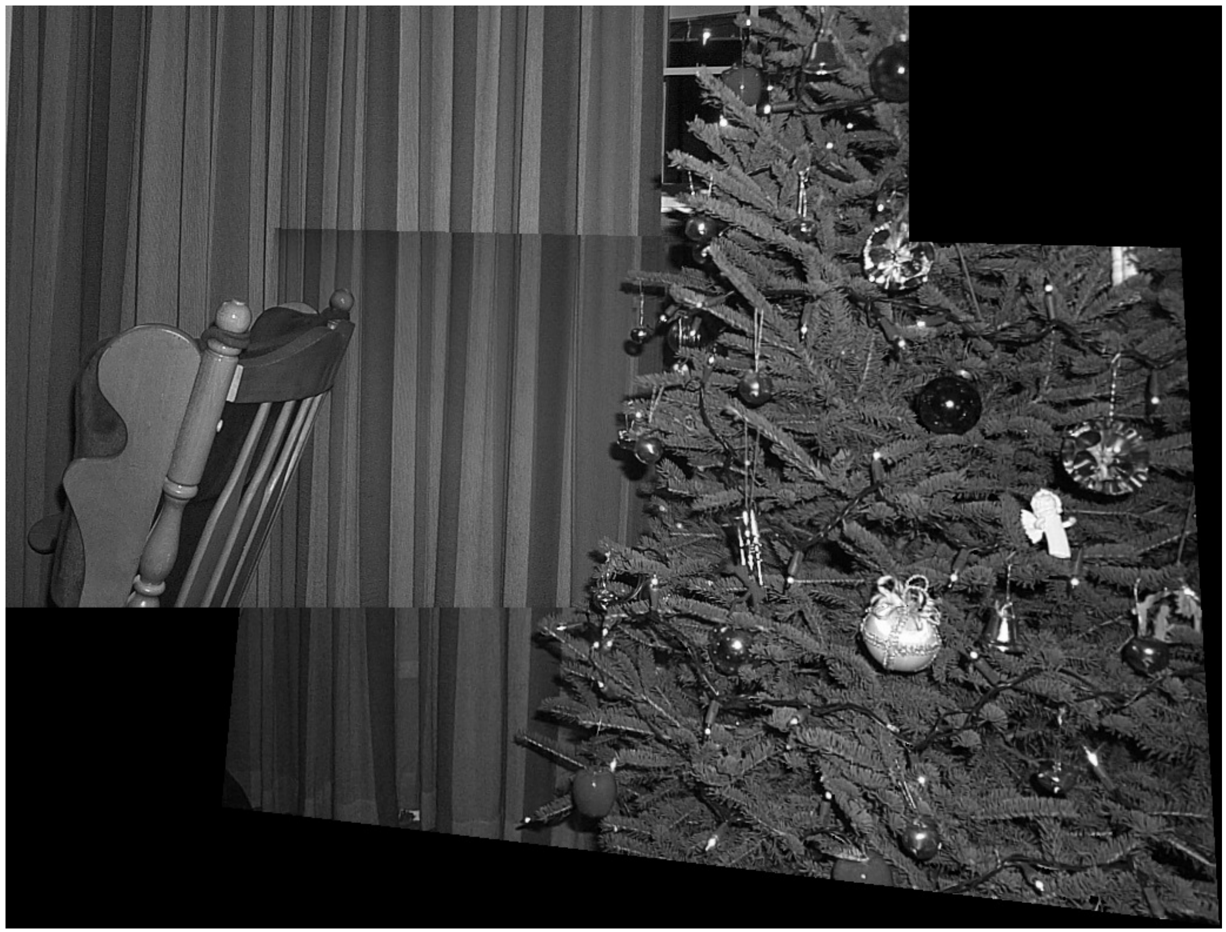
- Given a set of correspondences; pixels in left image that equal the right image
- Write down homography equations that must related these correpsondences x <-> x'
- Compute the homography using the same method as we used to compute fundamental matrix or to compute the projection matrix
- Basically compute the eigenvector assoicated with the smallest eigenvalue of the matrix A A^T

Automatic Mosaicing – Input

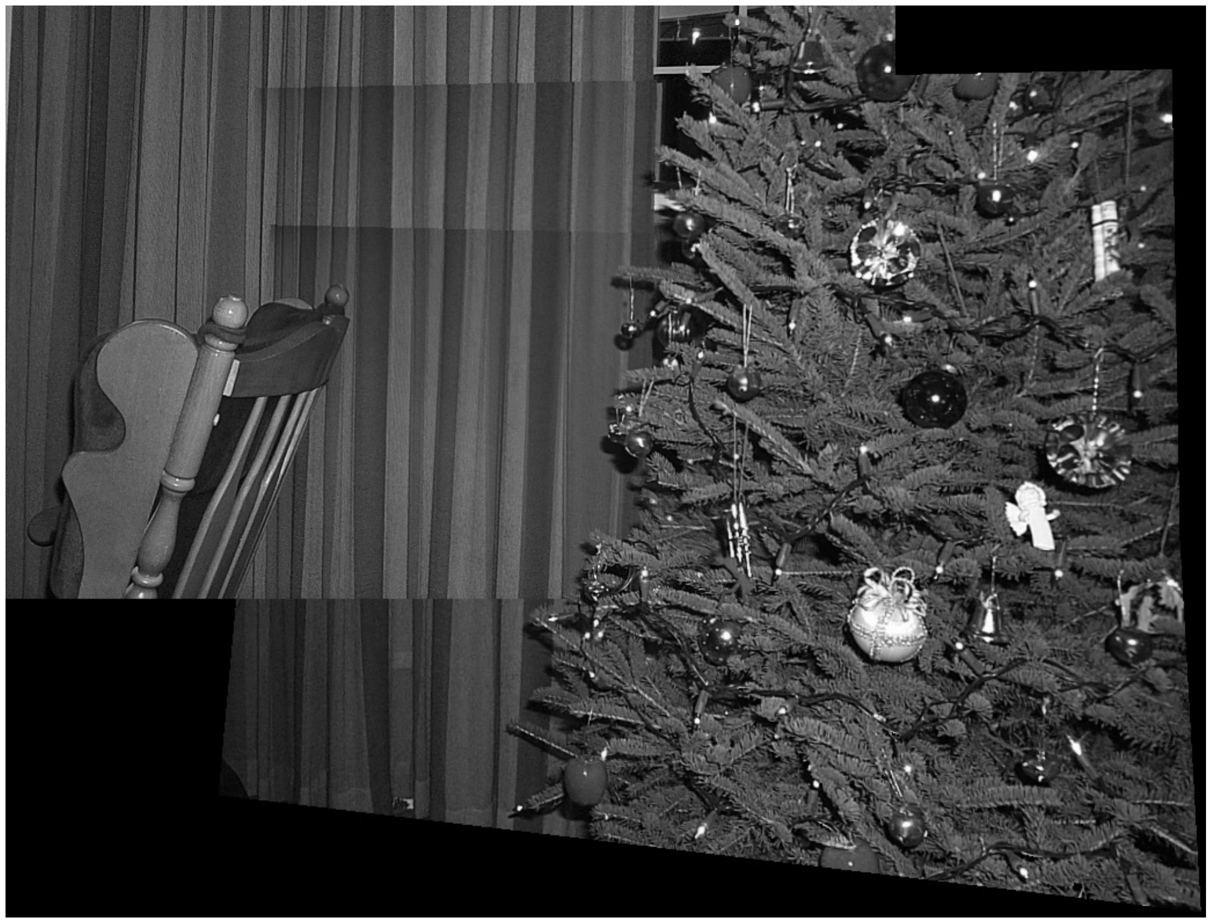
Automatic Mosaicing - Output



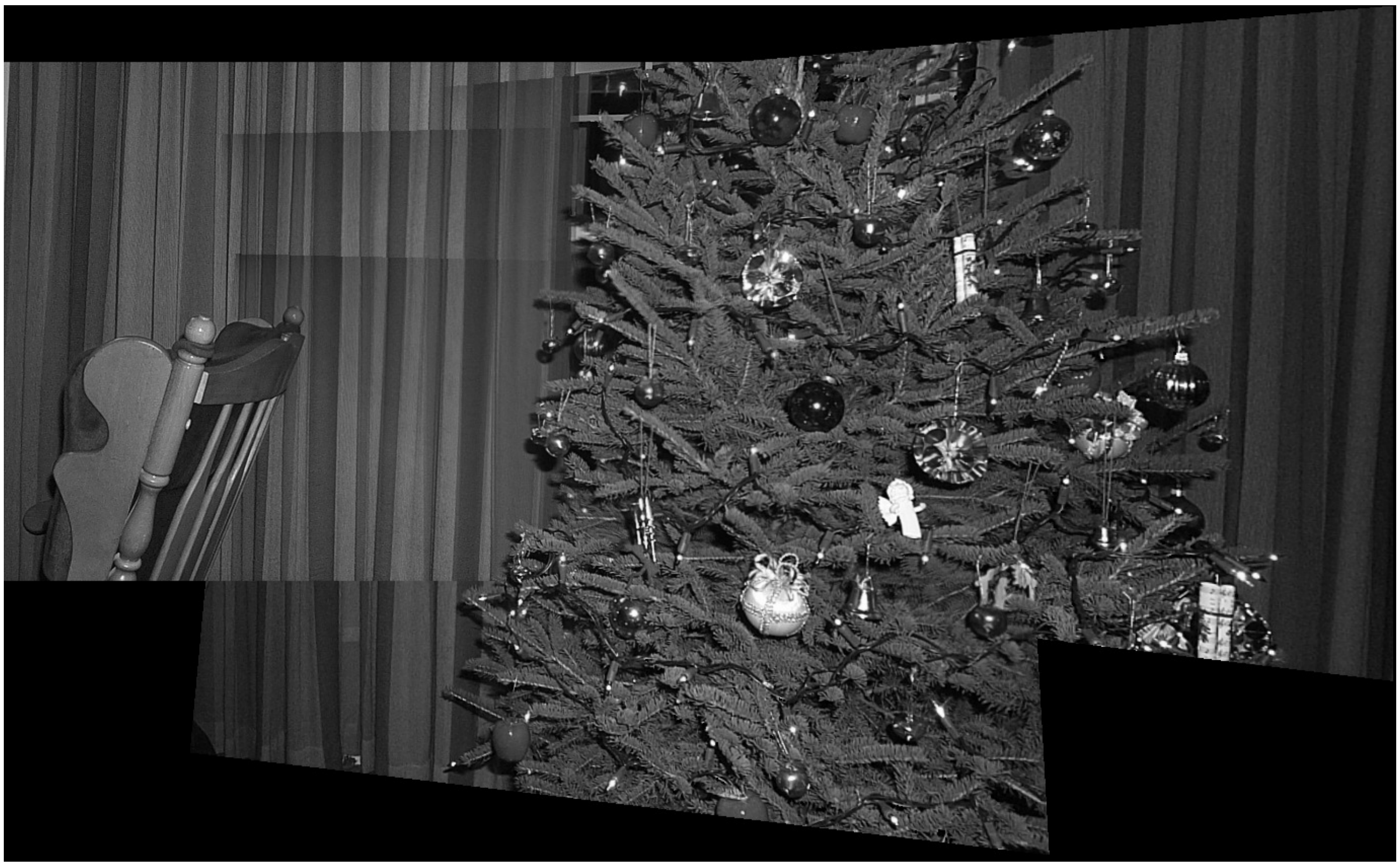
Automatic Mosaicing - Output



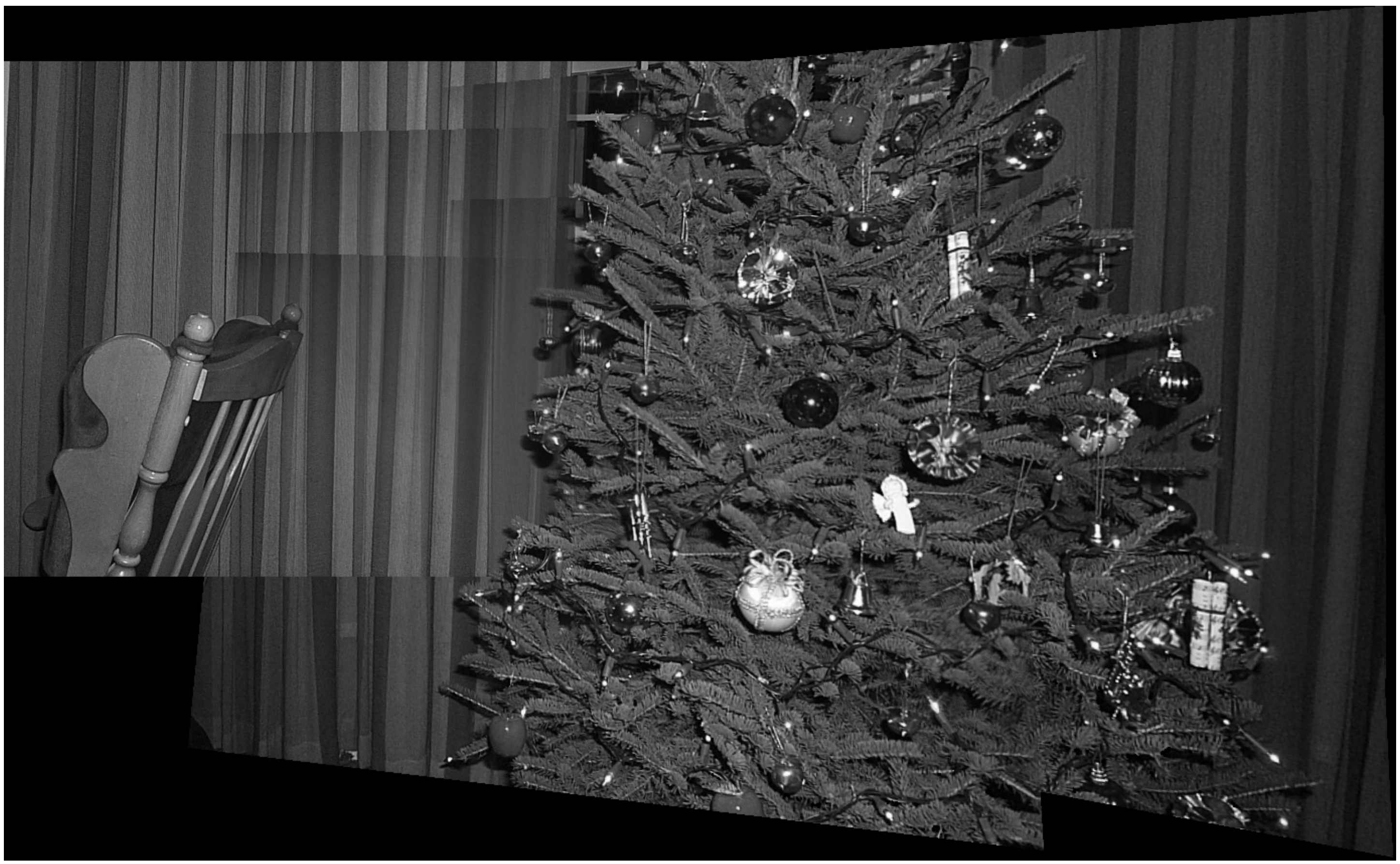
Automatic Mosaicing - Output



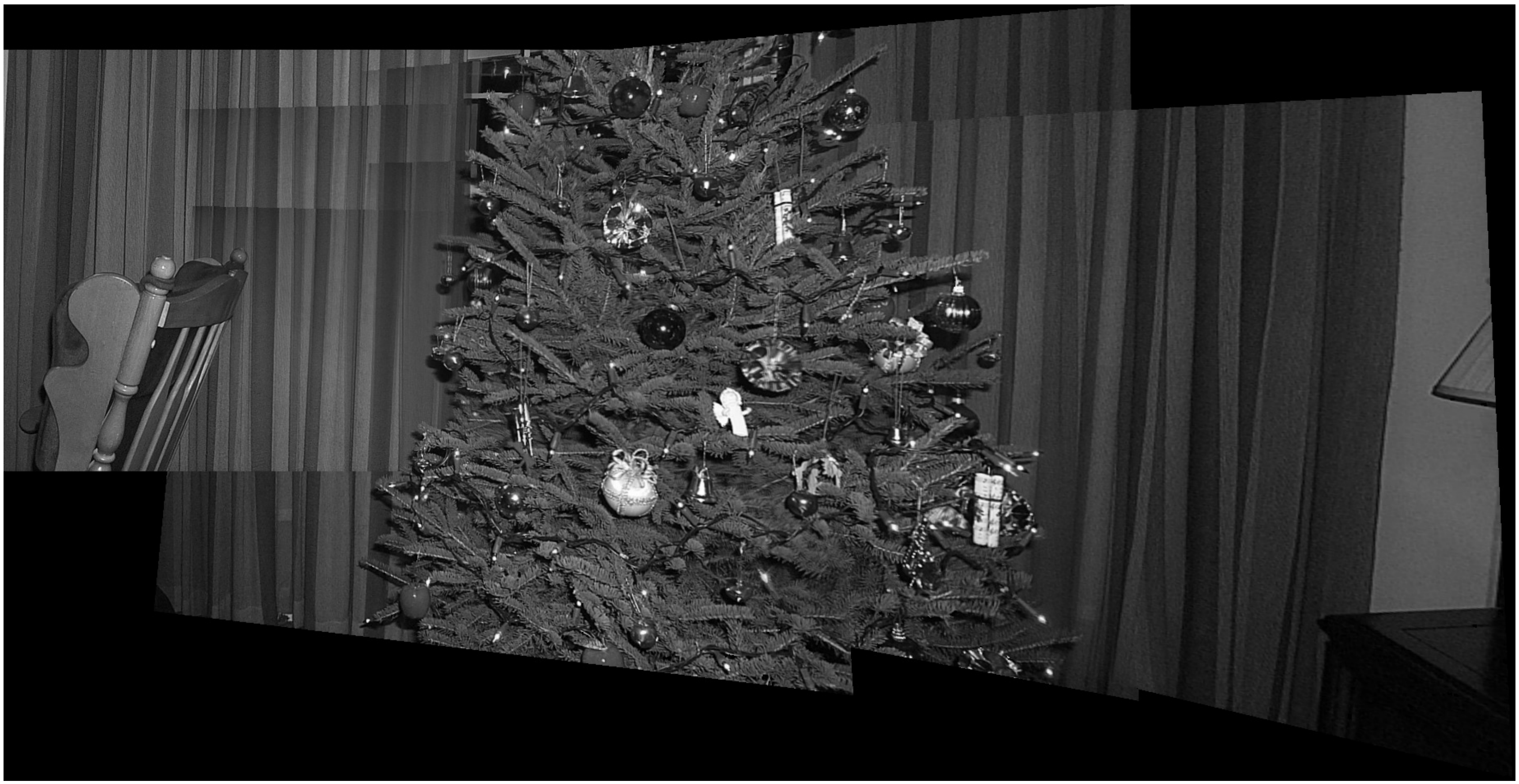
Automatic Mosaicing - Output



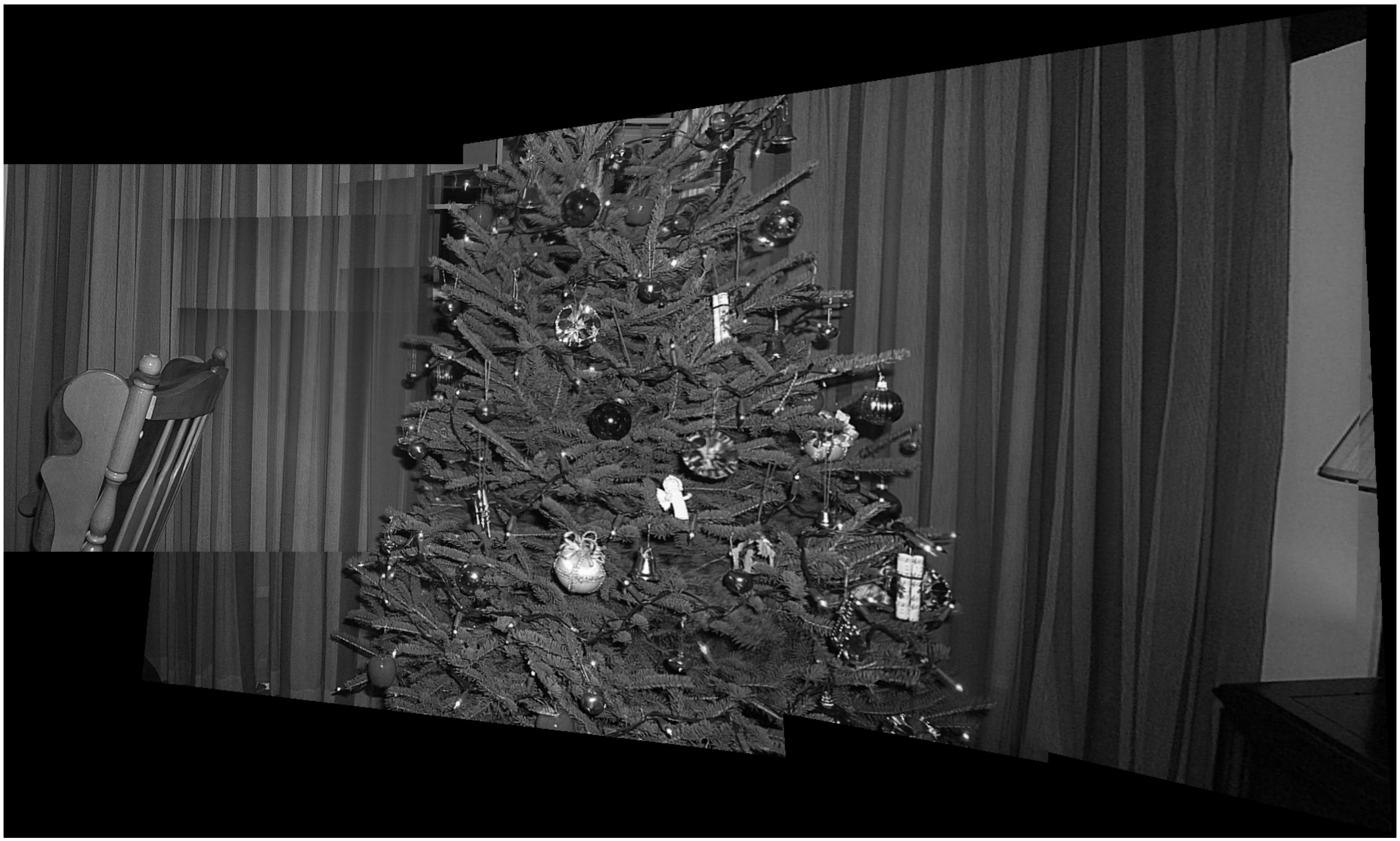
Automatic Mosaicing - Output



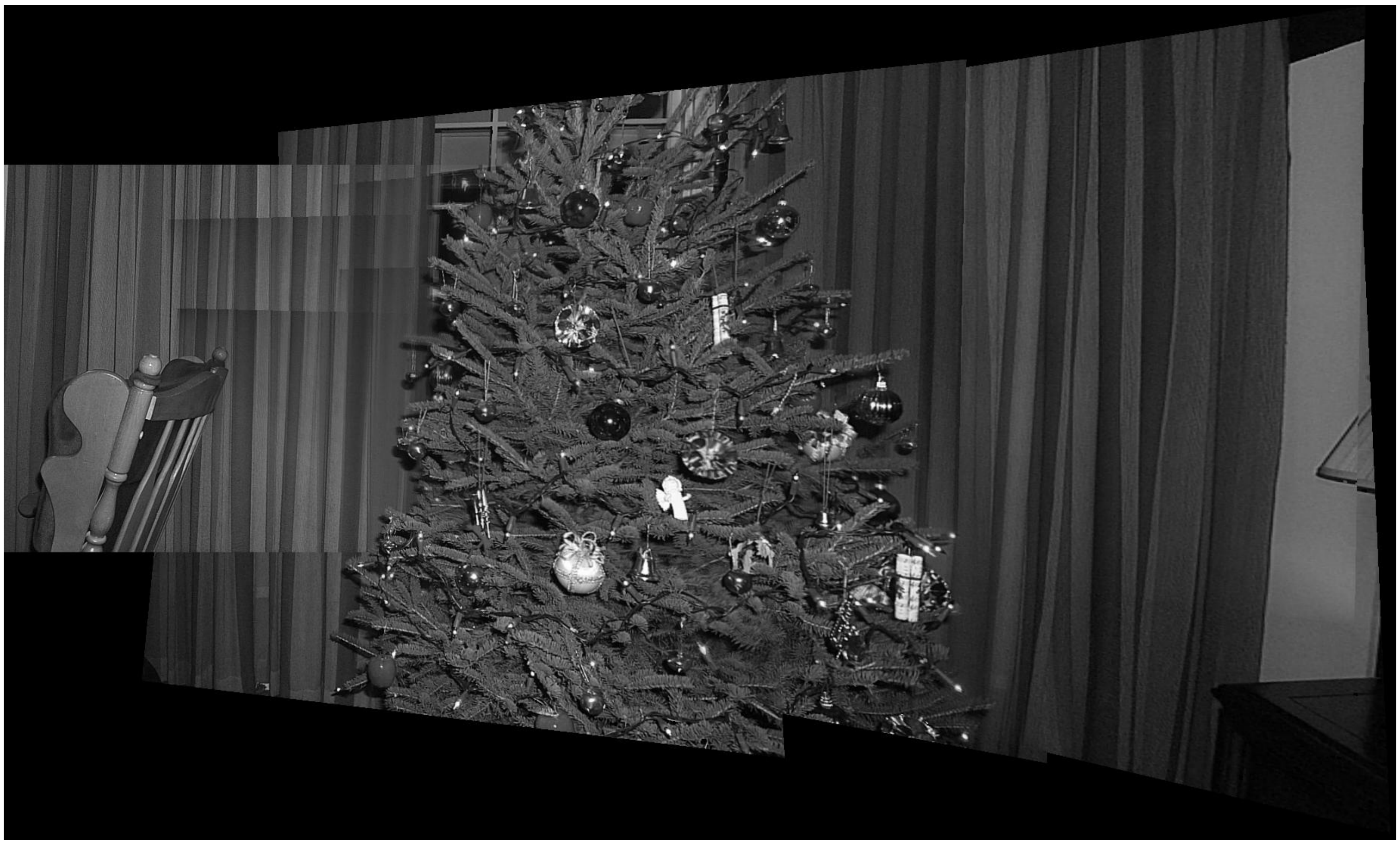
Automatic Mosaicing - Output



Automatic Mosaicing - Output



Automatic Mosaicing – Output



Computing mosaic

- Take a camera on a tripod
  - Rotate it around the axis of projection
  - Choose one of the images as the reference image
  - Compute the homography that aligns all the other images relative to that image
  - Need to find correspondences manually or automatically
- You get a single large image, a mosaic which looks like a high resolution image taken from that reference image
  - This large image is called a mosaic
- This is the basis of Quicktime mosaics

Computing homography

- For your purposes important to understand
  - When translation is null epipolar geometry fails to hold
  - In this case you only have image rotation
  - Can not compute depth for any point for which you have correspondences
  - Can compute the homography matrix from
    - The camera calibration and the know rotation or
    - Correspondences between the two images
  - If you just use correspondences you can make a mosaic from rotating images
  - Actually, can make a mosaic whenever you have a homograpy relationship
    - when you are looking at a plane and when rotating
    - Like having a high resolution camera from a single point of view