M4 - Lab 1: Image rectification

Sergio Montoya, Laia Albors, Ibrar Malik, Adam Szummer

Pompeu Fabra University
{ sergio.montoyadepaco, laia.albors, ibrar.malik, adam.szummer }@e-campus.uab.cat

1 Introduction

This week's goal is to learn about the many planar transformations that may be applied to an image and how to apply them. Another technique that will be learned is how to apply an affine and a metric rectification in order to remove the perspective distortion of an image. That is, to recover parallel lines and straight angles from an image that has been distorted by a projective transformation.

2 Homographies

A homography is a linear mapping in homogeneous coordinates such that $h : \mathbb{P}^2 \to \mathbb{P}^2$. When a homography is applied to a set of points, if they are part of a straight line, after the mapping they also belong to a straight line. A homography H is represented as a 3x3 non-singular matrix, which can be further classified into one of the different types of homographies.

2.1 Applying homographies

In order to correctly apply a homography H to an image, it has to be applied to each of the coordinates of the image. This results in a forward warping, which is not adequate as it can create holes in the transformed image. A better strategy is to perform an inverse warping, which can map each of the coordinates of the transformed image g into the original image f by applying the homography H^{-1}, hence avoiding the creation of holes. If a certain pixel falls into a non-integer location, the value can be interpolated.

The first step is to know the minimum and maximum coordinates of the transformed image g, so that we can map all the values in the convex set of those coordinates back into the original image f. To do so, we can first map the corners of the original image f into g by applying H. Once we obtain where the corners of f fall in g, we can apply an inverse warping on each coordinate of the transformed image g. The coordinates to be mapped are based on the known minimum coordinates and the width and height of the transformed image g. All the coordinates in the rectangle enclosing the transformed image are mapped, even if they fall outside the image f so that we obtain a rectangular image. In case a pixel falls outside of f, we can simply fill it with black. This process is depicted in Figure 1.

2.2 Similarity transformation

Let x and x' be points in homogeneous coordinates, then a similarity transformation H_S is a homography of the following form:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = H_S \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} s \cdot cos(\theta) & -s \cdot sin(\theta) & t_x \\ s \cdot sin(\theta) & s \cdot cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1}$$

This transformation can rotate, translate, and isotropically scale points. In Figure 2, we can observe some examples of this type of transformation. For observing the effects of changing the translation vector, we disable the automatic computation of the image size, because otherwise the differences cannot be visually analyzed.

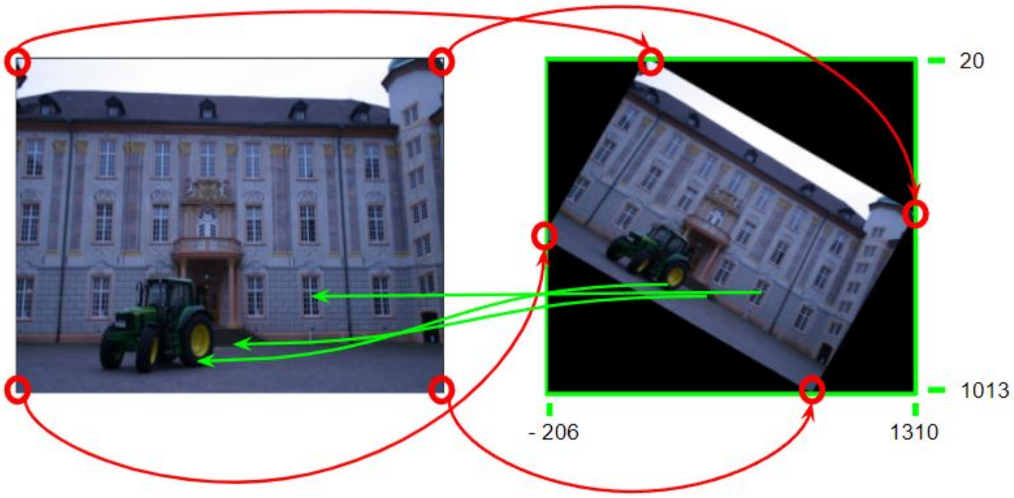


Fig. 1: Figure depicting the process of applying an homography to an image. (1) The first step consists of mapping the corners with a forward warping (denoted with red lines and circles). (2) After obtaining the minimum and maximum coordinates of each axis from the corner mapping, we can perform the inverse warping using all the coordinates enclosed between the minimum and maximum (green marks). For example, we can observe that the minimum x is -206 and the maximum x is 1310.

3 Affinity

An affine transformation is represented as

$$x' = H_A x = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} x \tag{2}$$

where A is a non-singular 2 x 2 matrix. A non-singular matrix is a matrix where the determinant is not zero hence $|ad - bc| \neq 0$. An affinity has 6 degrees of freedom.

3.1 Generate Matrix

The generation of an affine matrix is done by introducing a [2x2] matrix A:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \tag{3}$$

into the [3x3] homography matrix in the top left corner as shown in Equation 2.

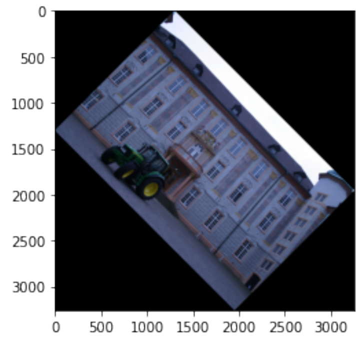
When applying matrix $A = \left[\begin{smallmatrix} 2 & 4 \\ 3 & 2 \end{smallmatrix}\right]$ the result is shown in Figure 3.

3.2 Decomposition of affinity

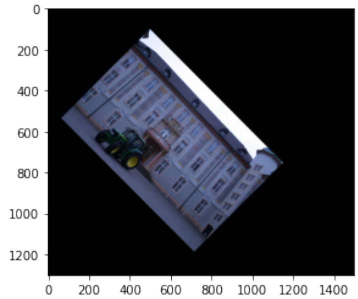
Matrix A can be decomposed into 4 different transformations:

- Rotation
- Scaling
- Rotation back to the original rotation.
- Final rotation.

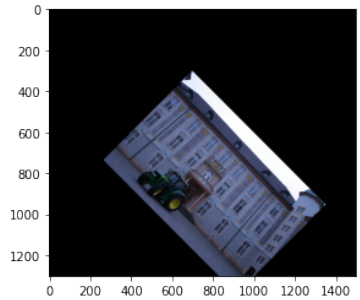
(a) Original image.



(b) Rotation by 45º and scaling of 3.



(c) Rotation by 45º and no translation.



(d) Rotation by 45º and translation of 200 pixels on each axis.

Fig. 2: Different similarity transformations. In figures (c) and (d), we remove the automatic size computation using the corners of the original image to be able to notice the translation visually.

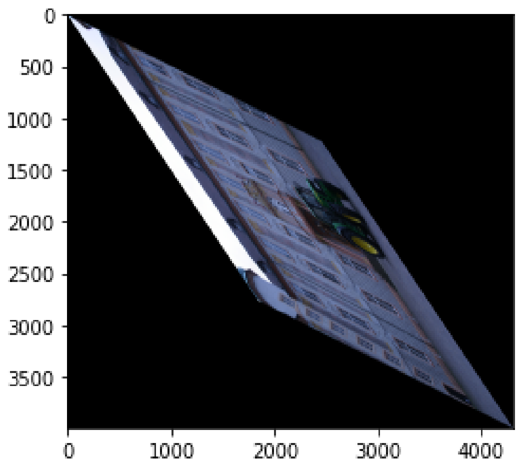


Fig. 3: Result of applying an affinity.

In matrix form, A is represented as

$$A = R(\theta)R(-\phi)DR(\phi) \tag{4}$$

$$\text{where } D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

To achieve this we use Singular Value Decomposition of A where $A = UDV^T = (UV^T)(VDV^T)$. Example for

$$A = \begin{bmatrix} 2 & 4 \\ 3 & 2 \end{bmatrix}$$

Decomposition using SVD:

$$U = \begin{bmatrix} -0.788 & -0.615 \\ -0.615 & 0.788 \end{bmatrix} \qquad D = \begin{bmatrix} 5.561 & 0 \\ 0 & 1.438 \end{bmatrix} \qquad V^T = \begin{bmatrix} -0.615 & -0.788 \\ 0.788 & -0.615 \end{bmatrix}$$

Finally, the first rotation $R(\theta) = UV^T$

$$UV^T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

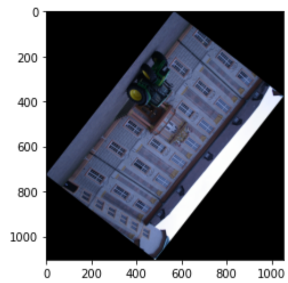
Second rotations $R(-\phi) = V \; R(\phi) = V^T$

$$V^T = \begin{bmatrix} -0.615 & -0.788 \\ 0.788 & -0.615 \end{bmatrix}$$

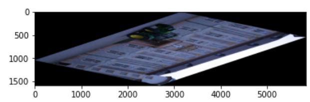
$$V = \begin{bmatrix} -0.615 & 0.788 \\ -0.788 & -0.615 \end{bmatrix}$$

3.3 Verification of products and separate transformations

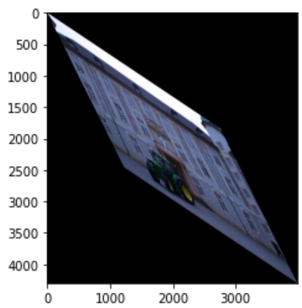
Verifying the product of separate steps proves to have the same value as the initial matrix A by using values from subsection 3.2 and Equation 4. The same can be proved by applying each transformation to the input image step by step and verifying the same result between Figure 3 and Figure 4d.



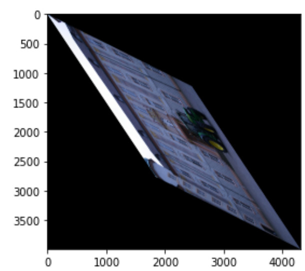
(a) Step 1 applying matrix V^T.



(b) Step 2 applying matrix D.



(c) Step 3 applying matrix V.



(d) Step 4 applying UV^T.

Fig. 4: Decomposed affinity steps.

4 Affine rectification

We can decompose any projective transformation into three components: the similarity H_S, affine H_A, and projective H_P components.

$$H = H_S H_A H_P = \begin{bmatrix} s\mathrm{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathrm{K} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & \mathbf{0} \\ \mathbf{v}^T & v \end{bmatrix} \tag{5}$$

Affine rectification will be the task of retrieving the H_P component so that at most an affine transformation has been applied to the image in the original plane.

This is done by observing a fact from affine transformations: they are the most general transformations that keep the line at infinity. We can see that H_P can map points at infinity $(x, y, 0)$ to any other point. The same can be shown for lines. Using this, we will search for the line at infinity in our transformed image, and find the linear mapping that moves this line into $l_\infty = (0, 0, 1)$.

All parallel lines intersect at a point in the line at infinity. If we have two pairs of lines that are parallel in the original plane, with their intersections we can define a line $l = (l_1, l_2, l_3)$. To map this line (and only this line) to $(0, 0, 1)$, we can use the following transform.

$$M = \begin{pmatrix} 1 & 0 & -l_1/l_3 \\ 0 & 1 & -l_2/l_3 \\ 0 & 0 & 1/l_3 \end{pmatrix} \tag{6}$$

We can see how $l_\infty = M \cdot l$. The corresponding dual transform for points will be $H_{\mathrm{AR}} = M^{-T}$. With this matrix we will be able to perform affine rectification.

$$H_{\mathrm{AR}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix} \tag{7}$$

4.1 Implementation

To remove the perspective distortion of an image we first have to identify some lines from it. However, in this lab we already have at our disposal the lines of each image, identified using LSD: a Line Segment Detector. These lines are given in a `txt` file, where each row has the coordinates of the first and last points of the corresponding line, among other information.

To apply affine rectification to an image, we have to identify 2 pairs of lines that are parallel in the real world, but aren't in the image. For the given images `0000_s.png` and `0001_s.png`, we used the lines shown in Figure 5a and Figure 5b, respectively.

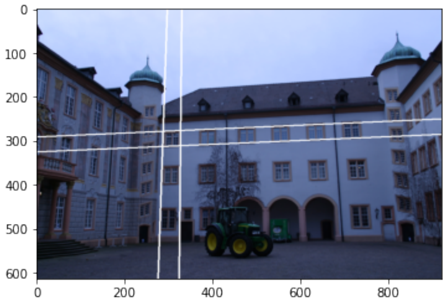
The next step is to find the vanishing line $l = (l_1, l_2, l_3)$ of the image. To do so, we have to compute the intersection points, p_{v1} and p_{v2}, of each pair of lines (by computing their cross product). The line that passes through these two points will be the vanishing line: $l = (l_1, l_2, l_3) = p_{v1} \times p_{v2}$.

With this line we can create the homography matrix that affinely rectifies the image (the one shown in Equation 7). The images resulting after applying affine rectification are shown in Figure 6a and Figure 6b.

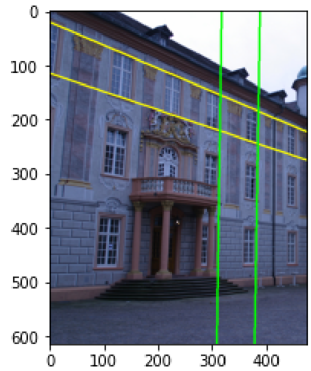
5 Metric rectification

In section 4, we performed an affine rectification, such that parallel lines now appear parallel in the image. The next step is to find the H_A matrix of Equation 5. When only a similarity is applied to an image, we say that the metric structure is conserved. Thus, metric rectification will be the task of recovering the image up to a similarity transformation.

Analogous to affine rectification, we will use the fact that the circular points are fixed under similarities but not under affine transformations. Instead of working directly with the circular points at infinity, we will

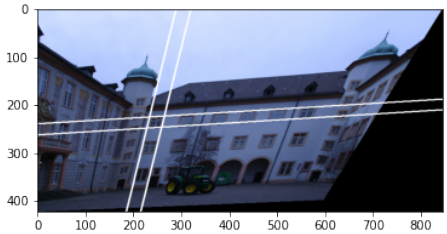


(a) `0000_s.png`

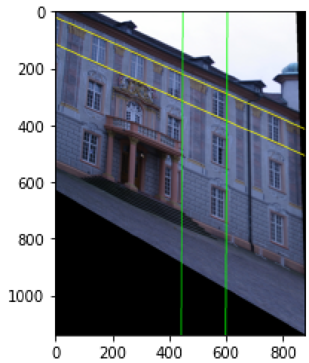


(b) `0001_s.png`

Fig. 5: Selected lines for each image to apply affine rectification.



(a) `0000_s.png`



(b) `0001_s.png`

Fig. 6: Affinely rectified images with the transformed lines.

work with the conic dual to these points, $C^*_\infty = IJ^T + JI^T = \mathrm{diag}(1, 1, 0)$. We do this because with C^*_∞ we can compute the angles between lines invariant to projective transforms, as follows:

$$\cos\theta = \frac{l^T C^*_\infty m}{\sqrt{(l^T C^*_\infty l)(m^T C^*_\infty m)}} \tag{8}$$

A dual conic is transformed like $C' = HCH^T$. By substituting this conic on Equation 8 and also transforming the lines $l' = H^{-T} l$ we can see that the angles remain constant on this measure.

From Equation 8, we know that orthogonal lines on the original plane should have a cosine equal to 0, or $l^T C^*_\infty m = 0$. And as this measure remains constant after projections, $l'^T C^{*\prime}_\infty m' = 0$. How does the conic C_∞ transform up to an affine transform?

$$C^{*\prime}_\infty = (H_A H_S) C^*_\infty (H_S^T H_A^T) = H_A C^*_\infty H_A^T = \begin{bmatrix} \mathrm{KK}^T & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \tag{9}$$

Using the definitions from Equation 5 and the matrix form of C^*_∞ it is easy to see how the circular points and its dual conic remain fixed under a similarity. Then, the conic is transformed as in Equation 9, and we do not care about H_P as we assume that affine rectification has been performed.

Our constraint is then $l^{T\prime} C^{*\prime}_\infty m' = 0$, where $C^{*\prime}_\infty$ is the matrix from Equation 9 and (l', m') is a pair of orthogonal lines after an affine transformation. We don't know the parameters of K, but as KK^T is a symmetric matrix we will only have 3 degrees of freedom. Luckily, it will suffice with 2 constraint equations as we do not care about the scale of this transformation. Solving this system of linear equations will give us the H_A transformation, and its inverse will metrically rectify the image.

5.1 Implementation

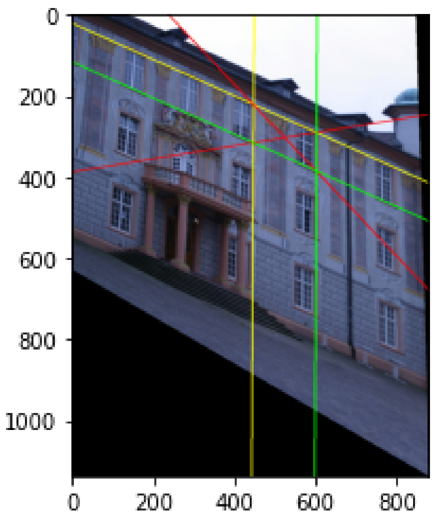
To apply metric rectification we will have to identify two pairs of orthogonal lines, which should be linearly independent between them. With $S = \mathrm{KK}^T$, we will reformulate the matrix at Equation 9, and write down $l^{T\prime} C_\infty^{*\prime} m' = 0$ as a single constraint:

$$l^{T\prime} C_\infty^{*\prime} m' = (l'_1 m'_1, l'_1 m'_2 + l'_2 m'_1, l'_2 m'_2)(s_{11}, s_{12}, s_{22})^T = 0 \tag{10}$$

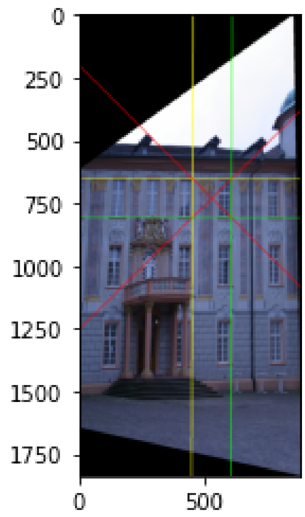
where s_{11}, s_{12} and s_{22} are the elements of S. This system needs 3 equations to be solved, but we can set any arbitrary scale by fixing one of the elements of S. We used the constraint of $s_{11} = 1$.

We will find the other 2 constraints of Equation 10 by finding two sets of orthogonal lines, and solving for s. To get the final K matrix, we just perform Cholesky decomposition on $S = \mathrm{KK}^T$. The final matrix to perform metric rectification will simply be $H_{MR} = H_A^{-1}$, built using K as in Equation 5. We can see this rectification in effect on Figure 7.

As we are performing metric rectification in two steps, its good to compose the H_{AR} and H_{MR} transformations in a single matrix. Because of the way we are applying homographies, by doing this in two steps we are both keeping already padded edges, and sampling from interpolated values instead of interpolating directly from the original ones.



(a) Affine rectified image



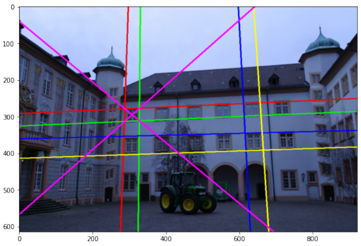
(b) Metric rectified image

Fig. 7: Metric rectified images with the transformed lines. We only use two pairs of lines.

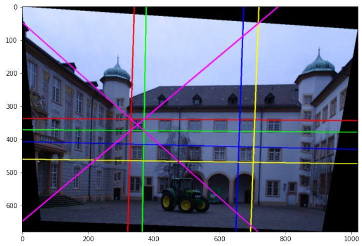
5.2 Optional task: Metric rectification in a single step

By identifying in the original image 5 pairs of perpendicular lines in the world, we can metrically rectify an image without performing first an affine rectification. Therefore, there is no need of identifying the transformed line at infinity l_∞. As the original image has not been affinely rectified first as in section 5, the conic dual to the circular points has been projectively transformed in the image. Hence, by knowing how dual conics transform under homographies, we have Equation 11.

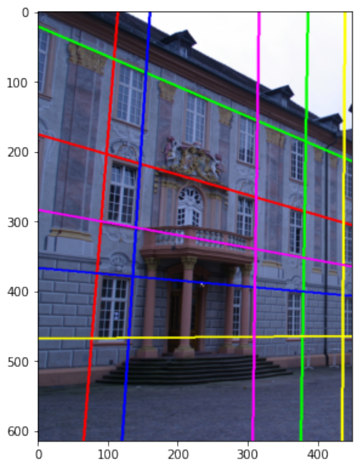
$$C_\infty^{*\prime} = (H_P H_A H_S) C_\infty^* (H_S^T H_A^T H_P^T) = H_P H_A C_\infty^* H_A^T H_P^T = \begin{bmatrix} \mathrm{KK}^T & \mathrm{KK}^T \mathbf{v} \\ \mathbf{v}^T \mathrm{KK}^T & \mathbf{v}^T \mathrm{KK}^T \mathbf{v} \end{bmatrix} \tag{11}$$



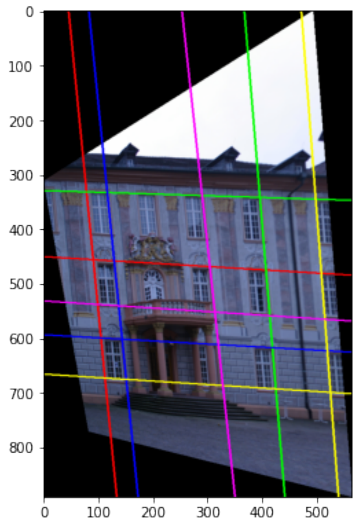
(a) Original image `0000_s.png`.



(b) Metrically rectified image `0000_s.png`.



(c) Original image `0001_s.png`.



(d) Metrically rectified image `0001_s.png`.

Fig. 8: Metric rectification in a single step. Pairs of orthogonal lines have the same color in images.

$C_\infty^{*\prime}$ can be determined by having 5 or more pairs of perpendicular lines. By knowing that these pairs satisfy that $l^T C_\infty^* m = 0$, we can rearrange these equations into a linear system and the conic is found as the solution to this linear system. Then, by decomposing $C_\infty^{*\prime}$ with SVD we have that:

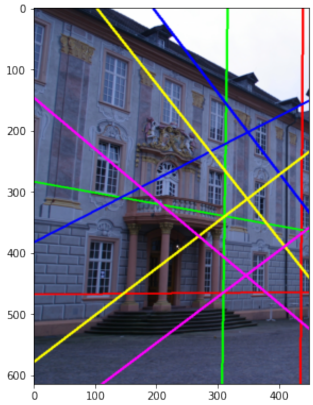
$$C_\infty^{*\prime} = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} U^T = U C_\infty^* U^T \tag{12}$$

By comparing Equation 11 and Equation 12, we can see that $H_P H_A = U$ (as dual conics transform as $C' = HCH^T$). Therefore, the metrically rectifying homography is $U^{-1} = U^T$. Two examples of metric rectification are provided in Figure 8. In this examples, we observe that the angles of the orthogonal lines after the metric rectification are not as close to orthogonal as with the stratified method. For example, in image `0001_s.png`, with the stratified method, angles are close to 90º, while with the single step rectification, with the best set of orthogonal lines that we have obtained, all angles are close to 81º. In image `0000_s.png`, we obtain results which are closer to the stratified method. We discuss in greater detail the differences between the two methods in section 6.

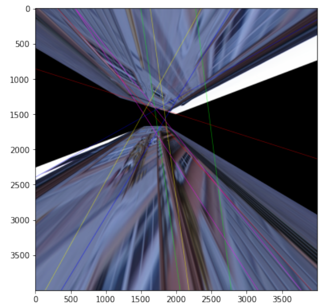
6 Discussion

6.1 Displaying lines correctly after applying an homography

To display an image after applying an homography H, we assume that the top left corner is the coordinate $(0, 0)^T$ of the image coordinate frame. Therefore, if we want to display a line in the image, we have to apply



(a) Original image.



(b) Wrong metrically rectified image.

Fig. 9: Metric rectification in a single step. These figures show how sensitive the method is to the choice of orthogonal lines.

the homography that moves the origin of the image to $(0,0)^T$. We can model such transformation with a similarity transformation T_n of the form:

$$T_n = \begin{pmatrix} 1 & 0 & -min_x \\ 0 & 1 & -min_y \\ 0 & 0 & 1 \end{pmatrix} \tag{13}$$

where min_x and min_y denote the minimum x and y coordinates. If we apply such transformation to lines we have that:

$$l' = T_n^{-T} l = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ min_x & min_y & 1 \end{pmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ y \\ x \cdot min_x + y \cdot min_y + z \end{bmatrix} \tag{14}$$

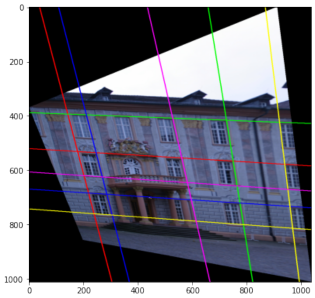
6.2 Stratified metric rectification vs metric rectification in one step

In this section, we compare the stratified metric rectification and the single step metric rectification. After rectifying the image with the single step metric rectification for different sets of orthogonal lines, we have observed that the method is very sensitive to the choice of the orthogonal lines and that it is not very robust. In Figure 9, we provide one example where the metric rectification provides very bad results just due to the selection of orthogonal lines.

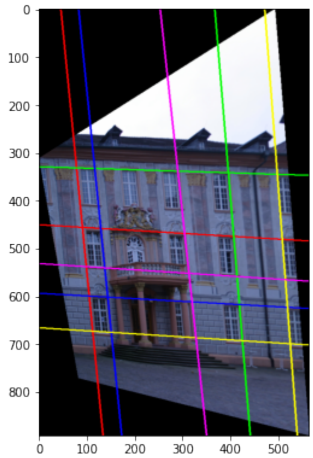
After applying the metric reconstruction in one step for different set of lines, we have been inspecting the SVD of $C^*_\infty = UDU^T$. We have observed that in all cases D is not equal to C^*_∞ (by inspecting Equation 12, D should be equal to the identity but with the last entry equal to 0). The closer D is to C^*_∞, the better the metric rectification will be. For example, if we compare the bad rectification from Figure 9 to a good reconstruction like Figure 8, when decomposing C^*_∞, we obtain the following values:

$$\text{Bad reconstruction}: D = \begin{bmatrix} 0.588 & 0 & 0 \\ 0 & 0.503 & 0 \\ 0 & 0 & 0.331 \end{bmatrix} \quad \text{Better reconstruction}: D = \begin{bmatrix} 0.855 & 0 & 0 \\ 0 & 0.433 & 0 \\ 0 & 0 & 0.0104 \end{bmatrix}$$

We have also noticed that the method for metric rectification in one step is not independent of the coordinate frame of the image, similar to the DLT algorithm. Specifically, if we transform points by a similarity



(a) Without normalization.



(b) With normalization.

Fig. 10: Metric rectification in a single step. Difference between (a) not using normalization and (b) using normalization. In (a) the mean angle is 74.2º and in (b) the mean angle is 80.8º.

transformation T, the resulting homography changes. Therefore, we have implemented a normalization technique as the one of the normalized DLT algorithm to minimize the effect of the image coordinate frame. First, we compute a transformation T so that the mean of the points is set to 0 and the mean distance to points is set to $\sqrt{2}$. Then, we proceed to compute the rectifying homography H with these normalized lines. After H is computed, we apply a similarity transformation to denormalize points, which is T^{-1}. This normalization technique does not only improve the results as can be seen in Figure 10, it also helps in avoiding numerical issues. When almost all the lines that we choose are almost linearly correlated, it is easy to run into numerical issues and the rank of the matrix A that is created from the constraints of orthogonal lines becomes less than 5. This problem becomes more problematic in cases of images like `0000_s.png`, where lines are very close to being linearly dependent as the projective distortion is not big. Without the normalization technique, we were unable to perform the rectification for numerical issues, but thanks to normalizing the coordinates all these numerical issues disappeared.

Even with the normalization technique, the method is very sensitive to the choice of lines and not very robust. To make sure that the problem was not regarding the non-orthogonality of the chosen lines in the world frame, we decide to compute the angle of all sets of orthogonal lines. We have applied the stratified rectification method and seen that almost all the sets of orthogonal lines are very close to 90º, discarding that possibility.

7 Conclusions

In this first assignment, we have seen how an affinity can be decomposed into four independent transformations, that can be applied separately to obtain the same result of applying the affinity. When constructing projective transformations, the last row of the homography matrix must be controlled, the h_{21} and h_{22} must be small to maintain the image structure. Regarding the metric rectification methods, we have explored the stratified and single step approaches. We have explored the importance of choosing a good set of lines. They must not be linearly dependent and must be correct (i.e. with the single step method, if the orthogonality of all pairs of lines is not realizable we will not get a correct rectification). After comparing both approaches, we have seen that the stratified rectification method is more robust than the single step rectification, which is very sensitive to the choice of orthogonal lines and not very robust. Finally, we have applied a normalization technique to improve the results on the single step rectification.