# C307 Lab-2: 2D Geometric Transforms

Submission Deadline: **Jan 24, 2022**

Through these lab exercises, we will learn how to perform some basic geometric transformations on the 2D plane. We will first use the familiar Cartesian coordinate system to do these transforms, and then move on to the homogeneous system.

## Exercise-1: Rotation using the familiar Cartesian system

In this exercise, we create a box and rotate it, all using the cartesian coordinate system. A partial MATLAB code is given in Fig 2. **Fill in the missing parts** in that code to get an output similar to Fig. 1 ( the exact degree of rotation need not be same. Suggested absolute degree value: 30)

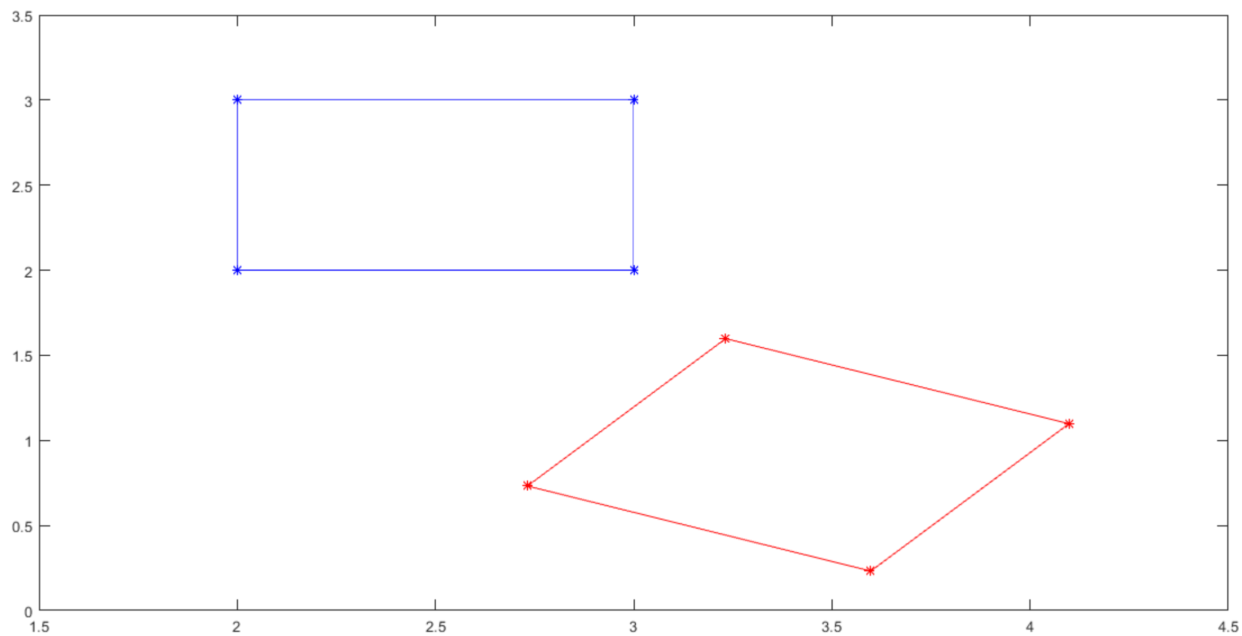



**Figure 1**: Rotating a box on a 2D plane; original (blue) and rotated (red).

```
% create box
% first row is horizontal coordinates; second row is vertical coordinates

my_pts = [2 2 3 3 2;2 3 3 2 2];

% write code here to display the original box
fig1=figure(1)


% write code here to create your 2D rotation matrix my_rot
```

# C307 Lab-2: 2D Geometric Transforms

```
my_rot = [];

% write code to perform rotation using my_rot and my_pts and store the result in my_rot_pts
my_rot_pts = [];

hold on;
%write code to plot output

axis([1.5 4.5 0 3.5]);
```

**Figure 2**: Partially completed code for Exercise-1.

## Exercise-2: Linear Translation using the Homogeneous system

In this exercise, we will perform a linear translation of a box; but, using the homogeneous system instead of the Cartesian. Remember, that for this to work, the box's coordinates need to be converted from Cartesian to homogeneous and the translation matrix also needs to be defined in the homogeneous system instead of the Cartesian. A partial MATLAB code is given in Fig 4. **Fill in the missing parts** in that code to get an output similar to Fig. 3 ( the exact magnitude of translation need not be the same ).

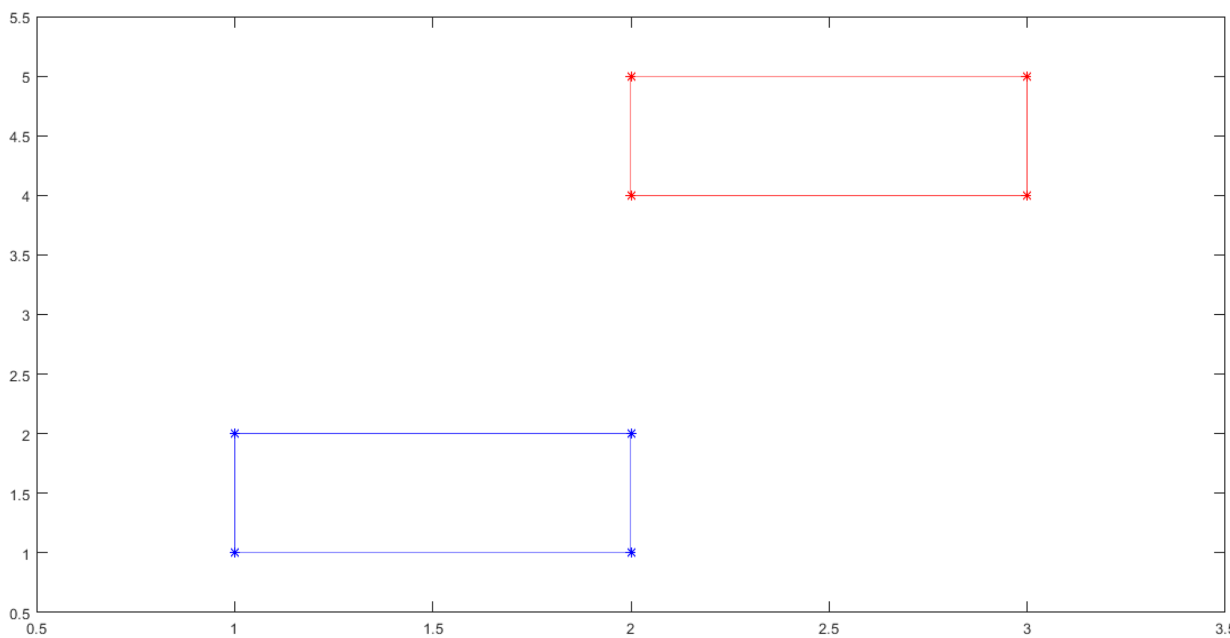



**Figure 3**: Translating a box on a 2D plane; original (blue) and rotated (red).

# C307 Lab-2: 2D Geometric Transforms

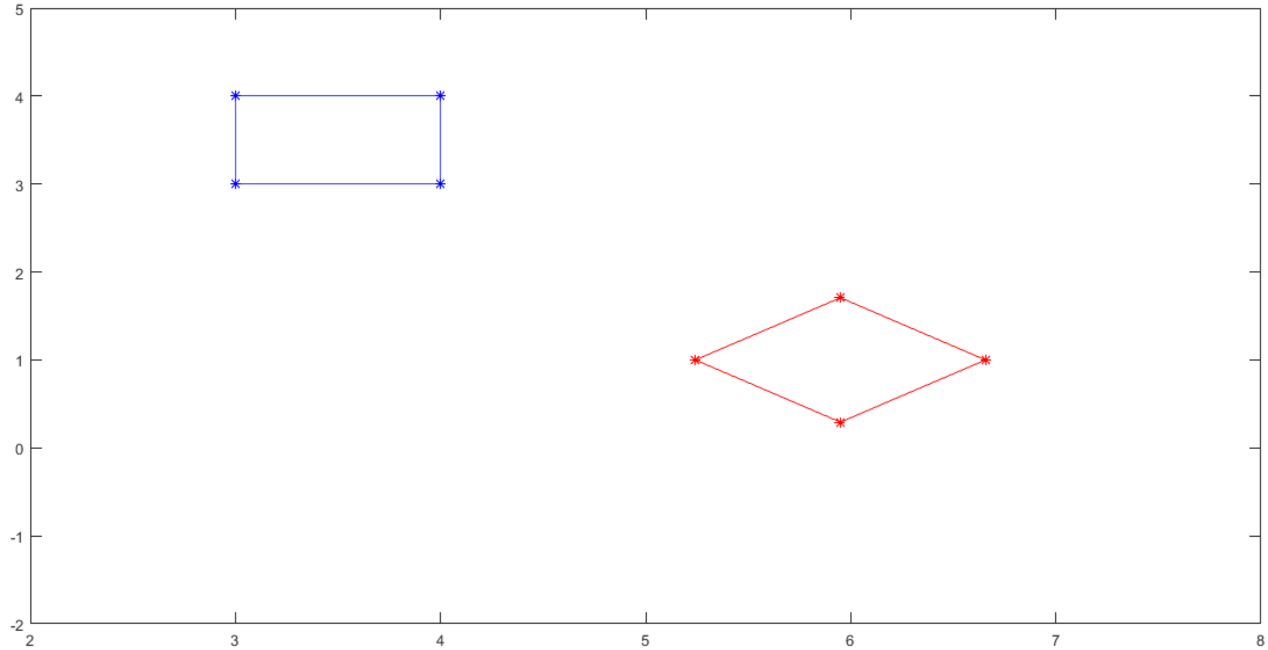
```
d_x = 1; % howmuchever you want to translate in horizontal direction
d_y = 3; % howmuchever you want to translate in vertical direction

% create the original box
% first row has horizontal coordinates, second row has vertical coordinates

my_points = [1 1 2 2 1;1 2 2 1 1];

% write code to plot the original box
fig1=figure(1)


% write code to create your Homogeneous 2D Translation matrix my_trans using d_x and d_y
my_trans = [];

% Next, we perform the translation

% write code to convert my_points to the homogeneous system and store the result in
hom_my_points
hom_my_points = [];

% write code to perform translation in the homogeneous system using my_trans and
hom_my_points and store the result in trans_my_points
trans_my_points = [];

hold on
% write code to plot the Translated box (output) which has to be done in Cartesian, so...
% cut out the X, Y points and ignore the 3rd dimension


axis([0.5 3.5 0.5 5.5]);    % just to make the plot nicely visible
```

**Figure 4**: Partially completed code for Exercise-2.

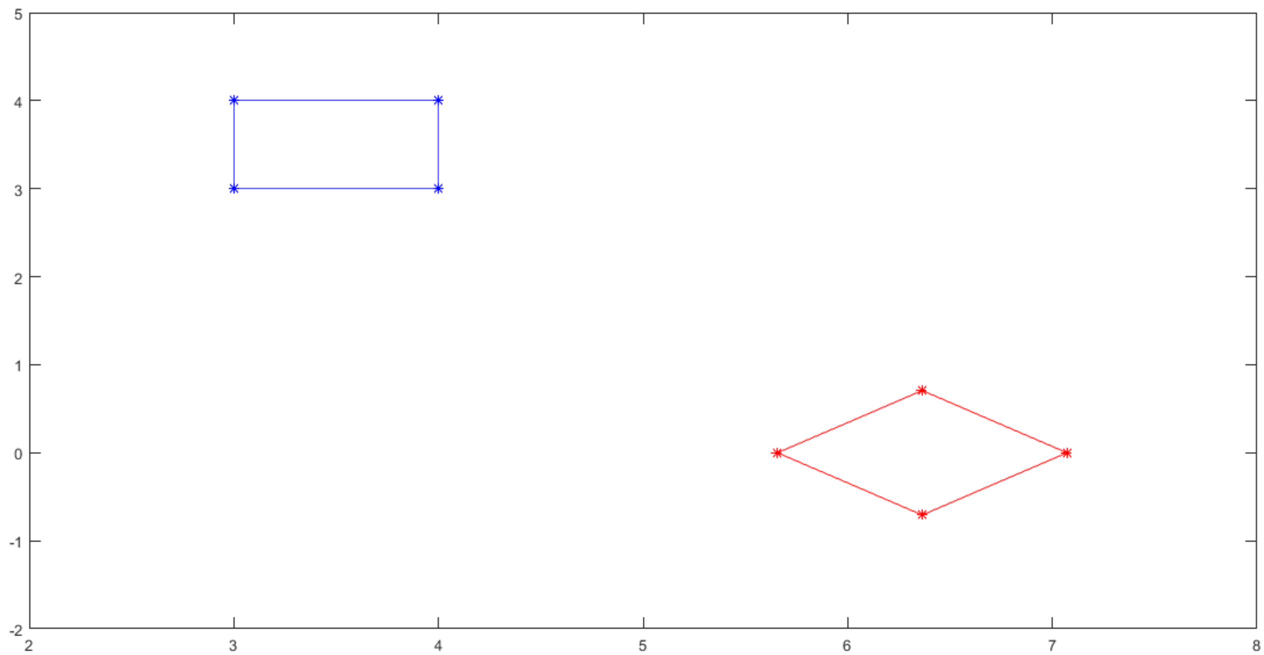
## Exercise-3: Compound transformation using the Homogeneous system

In this exercise, we will perform a combination of rotation and translation of a box as a single, "compound" transform using the homogeneous system. Then, we will reverse the order of the transforms. A partial MATLAB code is given in Fig 6. **Fill in the missing parts** in that code to get an output similar to Fig. 5 ( exact magnitudes of rotation and translation may differ).
Suggested absolute degree value: 45.

# C307 Lab-2: 2D Geometric Transforms




**(a) Rotation first then do translation in a single compound transform**




**(b) Translation first then do rotation in a single compound transform**

**Figure 5**: Compound transforms of a box on a 2D plane; original (blue) and rotated (red).

```
% we will perform a rotation followed by a translation, and then in reverse
% order and compare the results, both using the homogeneous system

d_x = 1; % howmuchever you want to translate in horizontal direction
```

# C307 Lab-2: 2D Geometric Transforms

```
d_y = 1; % howmuchever you want to translate in vertical direction

% create original box
% first row is horizontal and second row is vertical coordinates

my_pts = [3 3 4 4 3;3 4 4 3 3];

% write code to plot the box
fig1=figure(1)


% write code here to create your 2D rotation matrix my_rot
my_rot = [];

% write code to create your Homogeneous 2D Translation matrix hom_trans using d_x & d_y
hom_trans = [];

% Perform Compound transformation

% write code to construct your 2D Homogeneous Rotation Matrix using my_rot and store the
result in hom_rot
% HINT: start with a 3x3 identity matrix and replace a part of it with my_rot to create hom_rot
hom_rot = [];

% write code to convert my_pts to the homogeneous system and store the result in
hom_my_pts
hom_my_pts = [];

% write code to perform in a single compound transformation: translation (hom_trans)
followed by rotation (hom_rot) on hom_my_pts, and store the result in trans_my_pts
trans_my_pts = [];

% write code to plot the transformed box (output) which has to be done in Cartesian, so...
% cut out the X, Y points and ignore the 3rd dimension

hold on

axis([2 8 -2 5]);       % just to make the plot nicely visible

% Now, let us reverse the order of rotation and translation and compare

fig2=figure(2);

plot(my_pts(1,1:end),my_pts(2,1:end),'b*-');

% write code to perform in a single compound transformation: rotation followed by translation,
and store the result in trans_my_pts
```

```
trans_my_pts = [];

% write code to plot the Transformed box (output) which has to be done in Cartesian, so...
% cut out the X, Y points and ignore the 3rd dimension

hold on

axis([2 8 -2 5]);       % just to make the plot nicely visible
```

**Figure 6**: Partially completed code for Exercise-3.