Lab 4: Image Registration

cpselect()

```
base = imread('fish
floating = imread('
floating = floating

cpselect(floating,b
```

cpcorr()

```
movingPointsAdjusted = cpcorr(movingPoints,fixedPoints,floating,base)
```


fitgeotans()

```
tform = fitgeotrans(movingPointsAdjusted,fixedPoints,'affine');
```


imwarp()

```
floatingRegistered = imwarp(floating,tform,'FillValues',0,'OutputView',imref2d(size(base)));
figure
imshowpair(base,floatingRegistered)
```


Task 1: What is the effect of increasing/decreasing the number of chosen control points in registration accuracy?

We evaluate the effect of increasing/decreasing the number of control points by using methods described in ***Task 2***. We can conclude that increasing the number of control points had a positive impact in the localisation of the object of interest. Each control acts as a feature for the transformation, increasing the number of important features will provide more information for the transformation and thus increases its likelihood of success. However, it seems the effect of increasing the number of control points has a saturation point. Not only is there a saturation point but the additional control points need to be relevant, corresponding to key features in th ... hers.

It should also be not ... the control points had a much greater effect on the accuracy than the n ... nts. We were able to achieve results where more control points result ... on if the placement was suboptimal, even when using the cpcorr() fu ... the placement.

Another advantage of increasing the number of control points is that it increases the number of degrees of freedom that can be calculated. For simple transformations, such as translation, rotations and scaling (Nonreflective Similarity), only two control points are needed to calculate the given transformation. For more advanced transformations more control points will be needed such as a minimum of 3 for Affine and 4 for Projective.

Task 2: How would you evaluate the accuracy of your registration?

A number of different approaches can be taken to achieve this task, with the most simplest being qualitative evaluation of the registered images, simply observing the overlaid images.

However for a more automated and quantitative approach other methods can be explored. First we explore the use of intensity based methods.

The two images were of different modes which meant that intensity based methods produced poor results however the implementation was completed:

Root Mean Squared Error (RMSE) calculates the square root of the average squared difference between the predicted pixel values and the actual pixel values. The lower the RMSE, the better a model fits a dataset.

```
original_rmse = sqrt(immse(255-base,floating)) // 50.103
registered_rmse = sqrt(immse(255-base,floatingRegistered)) //51.53
```

Here the intensity of the base image is inverted to align with the floating image. As can be seen when registered there is a reduction in the RMSE value however not by a statistically significant amount especially considering how the RMSE value can be affected by noise and differences in the background of the image.

Another method for evaluating the registration prediction is using **Structural Similarity Index Measure (SSIM)**, SSIM is a perception-based metric that considers image degradation as perceived change in structural information, while also incorporating important perceptual phenomena, including both luminance masking and contrast masking terms. The difference with other techniques such as RMSE is that these approaches estimate absolute errors. Structural information is the idea that the pixels have strong inter-dependencies especially when they are spatially close. These dependencies carry important information about the structure of the objects in the visual scene. Implementation in Matlab produces a SSIM map as well as the overall score between 0 (poor SSIM) and 1 (perfect SSIM).

```
[ssimval,ssimmap] = ssim(registered,fishgt_img);
imshow(ssimmap,[]
title("Local SSIM [illegible]M Value: "+num2str(ssimval))
```

One drawback of bo[illegible]that they are influenced by the background which could mean they are[illegible] of the image registration task as a proportion of background pixels t[illegible]mation and would be categorised as being perfectly transformed.

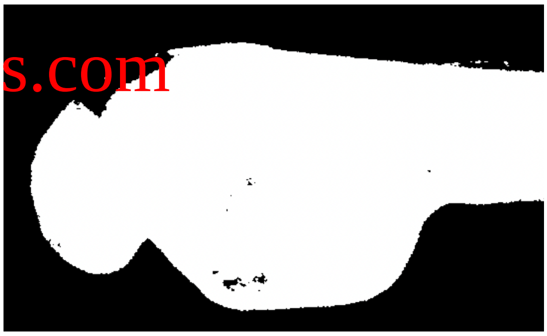
Feature based methods as opposed to intensity based methods, work by comparing the alignment of features between two images, such as control points which were not used to calculate the transformation, alignment of edges or segmentation.

Below we evaluate the accuracy of the registration through segmentation. Both the images are thresholded to extract the shape of the object. The two segmentations are then compared using the jaccard() function to evaluate the similarity (higher is better).

```
seg_base = (255-base)>40;
seg_floating = floating>40;
seg_floatingRegistered = floatingRegistered>40;

jaccard(seg_base,seg_floatingRegistered)
jaccard(seg_base,seg_floating)
```

seg_base

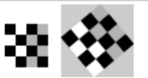


seg_floatingRegistered

No Registration	0.7757
3 Control Points	0.8227
4 Control Points	0.8677
5 Control Points	0.8800

Task 3: Other than Affine, what are the other options and which one do you think works best?

A list of options can be found here https://uk.mathworks.com/help/images/ref/fitgeotrans.html in the **More About > Transformation Types** section. This section outlines the types of transformations and where you would use each of these types.

Transformation Type	[illegible]	Minimum Number of Control Point Pairs	Example
'nonreflectivesimilarity'	[illegible] s in the moving image are unchanged, but the image [illegible] translation, rotation, and scaling. Straight lines [illegible] still parallel.	2	
'similarity'	[illegible] rity' with the addition of optional reflection.	3	
'affine'	Use this transformation when shapes in the moving image exhibit shearing. Straight lines remain straight, and parallel lines remain parallel, but rectangles become parallelograms.	3	
'projective'	Use this transformation when the scene appears tilted. Straight lines remain straight, but parallel lines converge toward a vanishing point.	4	
'polynomial'	Use this transformation when objects in the image are curved. The higher the order of the polynomial, the better the fit, but the result can contain more curves than the fixed image.	6 (order 2) 10 (order 3) 15 (order 4)	
'pwl'	Use this transformation (piecewise linear) when parts of the image appear distorted differently.	4	
'lwm'	Use this transformation (local weighted mean) when the distortion varies locally and piecewise linear is not sufficient.	6 (12 recommended)	

Using affine as a baseline we note similarity and non-reflective similarity generated a very small decrease in RMSE which suggests that there has perhaps been some combination of translation, rotation and scaling used in the images. However, the SSIM of affine remained higher. No difference shown using projective while pw1, 2,3,4 degree polynomial decrease performance suggesting that the image has been distorted as a whole rather than different distortions across different patches of the image.