```python
img_vis = io.imread('FishImageFiles/fish-vis.tif')
img_cfp = io.imread('FishImageFiles/fish-cfp-1.tif')
```

Lab 4: Image Registration

Q1:

Using more control points generally improves the accuracy of the registration. The additional control points provide more correspondences between images, allowing the registration algorithm to estimate the transformation between images more accurately. With more control points, the algorithm can use more information to optimize the transformation and thus obtain more accurate alignment results.

On the contrary, **reducing the number of control points** may reduce the accuracy of the registration. Reducing control points reduces the amount of information processed by the algorithm and may lead to less accurate estimation of the transformation. This may lead to misalignment or distortion of the registered image.

Overall, increasing the number of control points generally improves the registration accuracy because it provides more information to help the algorithm optimize the transformation. While reducing the number of control points may lead to lower accuracy because of the lack of sufficient information for accurate alignment.

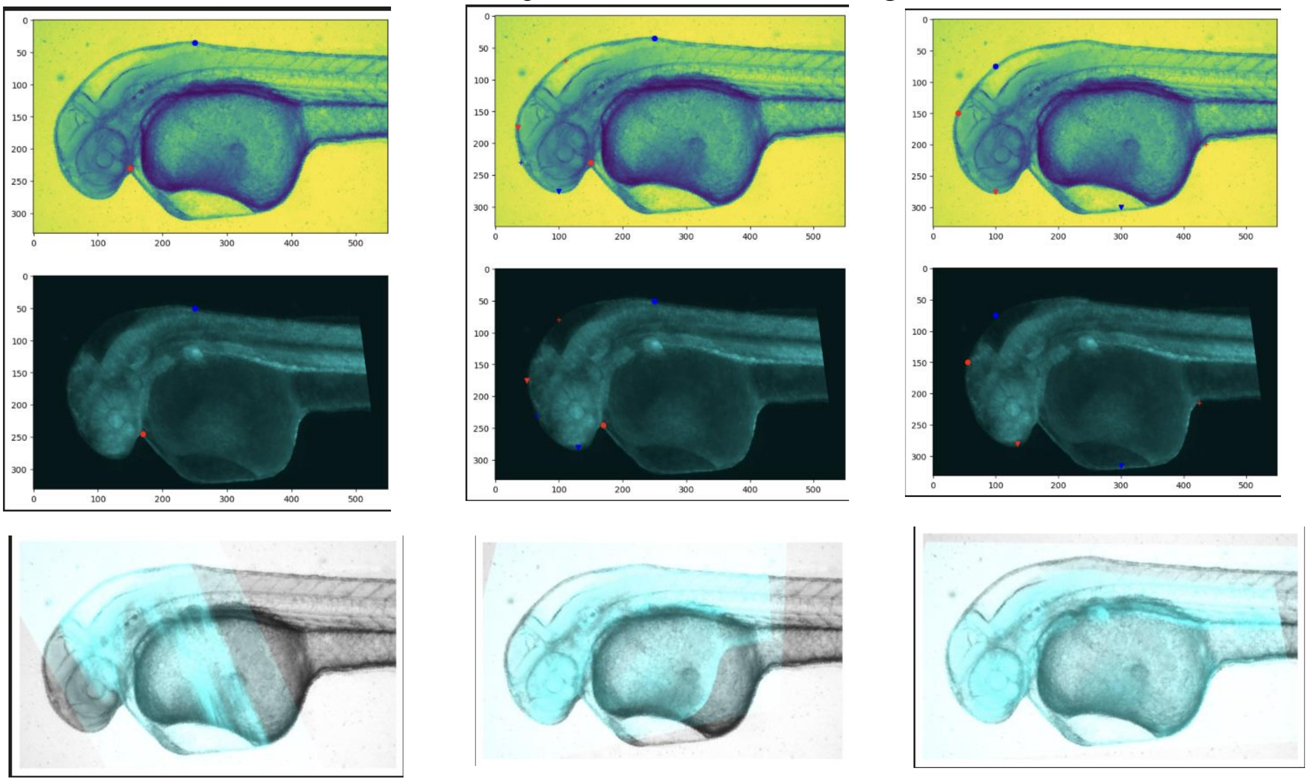
Using control points that average cover images is a common and effective strategy to improve the accuracy of image registration. This approach distributes control points over critical regions of the image to ensure a good estimation of the transformation of the whole image. Here are some advantages of this approach:

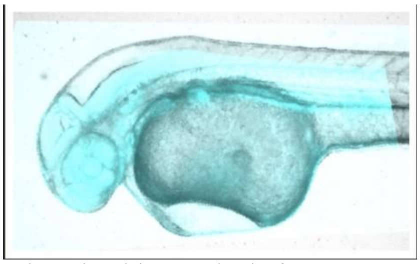
Globality: By distributing control points evenly over the entire image, accurate estimation of the transformation can be ensured for the entire image. This helps to avoid registration distortion due to the lack of control points in some regions.

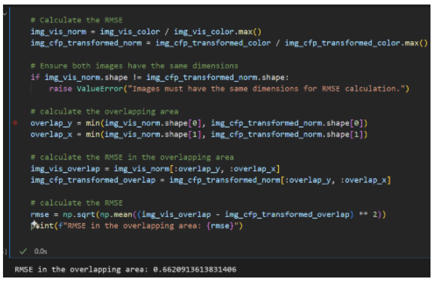
Robustness: Averaging the control points covering the image can improve the robustness of the algorithm as they are able to capture all aspects and features of the image. This helps to reduce registration errors due to insufficient local features or abnormal regions.

Representativity: The average coverage of the control points ensures that they represent the overall characteristics and structure of the image. In this way, the registration algorithm can gain a better understanding of the correspondence between the images and thus produce a more accurate estimate of the transformation.

From left to right are **decrease**, **increase** and **avg.**







```python
# Calculate the RMSE
img_vis_norm = img_vis_color / img_vis_color.max()
img_cfp_transformed_norm = img_cfp_transformed_color / img_cfp_transformed_color.max()

# Ensure both images have the same dimensions
if img_vis_norm.shape != img_cfp_transformed_norm.shape:
    raise ValueError("Images must have the same dimensions for RMSE calculation.")

# calculate the overlapping area
overlap_y = min(img_vis_norm.shape[0], img_cfp_transformed_norm.shape[0])
overlap_x = min(img_vis_norm.shape[1], img_cfp_transformed_norm.shape[1])

# calculate the RMSE in the overlapping area
img_vis_overlap = img_vis_norm[:overlap_y, :overlap_x]
img_cfp_transformed_overlap = img_cfp_transformed_norm[:overlap_y, :overlap_x]

# calculate the RMSE
rmse = np.sqrt(np.mean((img_vis_overlap - img_cfp_transformed_overlap) ** 2))
print(f"RMSE in the overlapping area: {rmse}")
```

```
0.0s
RMSE in the overlapping area: 0.6620913613831406
```

Q2:

1. **Visual Observation**: Visual observation is an intuitive method of assessment. It involves examining the overlaid images after registration to determine whether the two images are correctly aligned in space. This method is very effective in detecting major alignment errors, but it is subjective and not sensitive enough to minor registration inaccuracies.

2. **Calculating Root Mean Square Error (RMSE):** For registration tasks, RMSE is commonly used for evaluation. RMSE provides a quantitative way to assess the accuracy of registration. It calculates the average squared difference between corresponding pixels of the two images, offering objective information about the quality of image registration. However, before calculating RMSE, it is necessary to standardize the images if they have **different pixel value ranges**.
   Standardizing Images: Scale the pixel values of both images to the same range, typically from 0 to 1 This ensures that the RMSE calculation is not affected by different intensity ranges of the images.
   Calculating RMSE: Calculate the RMSE on the standardized images. This is done by squaring the differences of corresponding pixels of the two images, averaging them, and then taking the square root. The final RMSE value obtained is **0.67.**

Q3:

After using **Affine, Projective, Similarity, and Euclidean** transformations, I found that the performance of the affine transformation is the best. Compared to the other three, it can register areas that the other three cannot, and visually, it appears to be the correct pairing. The reason why the Affine Transform provides better registration results in some cases compared to the other three types of transformations (Projective, Similarity, and Euclidean) is mainly because it offers a good balance of flexibility and control in transformation. Here are some possible reasons:

More Degrees of Freedom: The affine transformation offers more degrees of freedom compared to the rigid and similarity transformations, as it includes not only rotation and translation but also scaling and shearing. This allows the affine transformation to better adapt to changes in scale and some shape deformations between images.

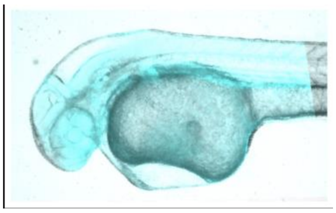
Preservation of Lines and Parallelism: Unlike projective transformations, affine transformations maintain the straightness of lines and the parallelism of lines.

Avoiding Excessive Deformation: Projective transformations introduce perspective distortions, which may cause excessive deformation in some images.

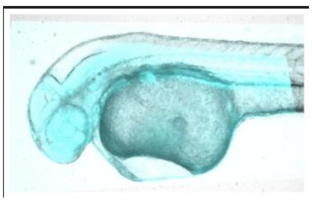
Adaptability: Due to its ability to handle various types of deformations, the affine transformation performs well in scenarios with minor perspective changes or when changes between images are not just limited to position and orientation.

Computational Efficiency: Compared to complex projective transformations and non-rigid transformations, affine transformations are generally easier to compute and optimize, which may lead to more stab le and accurate results in practical applications.

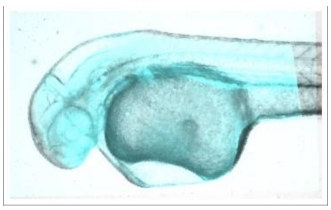
Similarity



Projective



Euclidean



Affine