

Signal Processing Assignment

5 (a) How to manually fix a target in the image in MATLAB

To manually fix a target in an image using MATLAB, you can use the "impoint" function. This function allows you to add a draggable point to an image, which you can then use to manually adjust the position of the target. Here is an example of how to use this function:

```
% Load the image into MATLAB
img = imread('myimage.png');

% Display the image
imshow(img);

% Add a draggable point to the image
h = impoint;

% Get the position of the point
p = h.getPosition;

% Use the position of the point to adjust the target in the image
% (exact implementation will depend on the specific details of your image and target)
targetX = p(1);
targetY = p(2);
```

Once you have added the draggable point and obtained its position, you can use that position to adjust the target in the image as needed. Again, the exact implementation will depend on the specific details of your image and target.

5 (b) Extract a patch of image containing the target and save it as the sample in MATLAB

To extract a patch of an image containing the target and save it as a sample in MATLAB, you can use the "imcrop" function. This function allows you to select a rectangular region of an image, which you can then save to a file using the "imwrite" function.

Here is an example of how to use these functions to extract a patch containing the target:

```
% Load the image into MATLAB
img = imread('myimage.png');

% Display the image
imshow(img);

% Use the imcrop function to select a rectangular region containing the target
patch = imcrop(img);

% Save the patch to a file
imwrite(patch, 'mypatch.png');
```

Once you have used the "imcrop" function to select a patch containing the target, you can save that patch to a file using the "imwrite" function. This will allow you to access the patch as a separate image that you can use for further processing or analysis.

5 (c) Add some noise to your original image in MATLAB

To add noise to an image in MATLAB, you can use the "imnoise" function. This function allows you to add various types of noise to an image, including Gaussian noise, salt-and-pepper noise, and speckle noise. Here is an example of how to use this function to add Gaussian noise to an image:

```
% Load the image into MATLAB
img = imread('myimage.png');

% Add Gaussian noise to the image
noisyImg = imnoise(img, 'gaussian');

% Display the noisy image
imshow(noisyImg);
```

This example adds Gaussian noise to the image, but you can also use the "imnoise" function to add other types of noise.

For example, you can use the following syntax to add salt-and-pepper noise to the image:

```
% Add salt-and-pepper noise to the image
noisyImg = imnoise(img, 'salt & pepper');
```

5 (d) Try to conceal a method to automatically detect the target (by using the sample) from the noisy image

To automatically detect the target in a noisy image, you can use a technique called template matching. This technique involves creating a "template" image that contains the target, and then using this template to search for the target in the noisy image.

In MATLAB, you can use the "normxcorr2" function to perform template matching. Here is an example of how to use this function to detect the target in a noisy image:

```
% Load the noisy image and the template image
img = imread('mynoisyimage.png');
template = imread('mytemplate.png');

% Use the normxcorr2 function to perform template matching
c = normxcorr2(template, img);

% Find the maximum value in the correlation matrix, which indicates the location of the target
[max_c, imax] = max(abs(c(:)));
[ypeak, xpeak] = ind2sub(size(c),imax(1));

% Adjust the coordinates of the target to account for the padding added by normxcorr2
corr_offset = [(xpeak-size(template,2)) (ypeak-size(template,1))];

% Display the detected target in the noisy image
figure;
imshow(img);
hold on;
rectangle('Position', [corr_offset(1)+1 corr_offset(2)+1 size(template,2) size(template,1)],
```

```
'EdgeColor', 'g', 'LineWidth', 2);
```

5 (e) Do more detection experiments by altering the level of noise or the type of noise

To perform more detection experiments by altering the level of noise or the type of noise, you can modify the parameters of the "imnoise" function that you used to add noise to the image. For example, you can use the following syntax to add Gaussian noise with a standard deviation of 0.5 and a mean of 0 to the image:

```
% Add Gaussian noise with a standard deviation of 0.5 and a mean of 0 to the image  
noisyImg = imnoise(img, 'gaussian', 0, 0.5);
```

Additionally, you can use the "imnoise" function to add other types of noise to the image, such as salt-and-pepper noise or speckle noise. For example, you can use the following syntax to add salt-and-pepper noise with a density of 0.05 to the image:

```
% Add salt-and-pepper noise with a density of 0.05 to the image  
noisyImg = imnoise(img, 'salt & pepper', 0.05);
```