
Table of Contents

Load Videos	1
Identify Initial Tracking Point	1
Manual adjustments to optimize point tracking results	2
Create x and y coordinate vectors tracking the top of the paint can through each video	3
Play videos with point tracking	3
Subtract off the mean, align vectors, and truncate x and y coordinate vectors	4
Oh the SVD won't let me be	4
Plot singular values to identify principal components	5
CASE 1 specific plots	6
CASE 2 specific plots	7
CASE 3 specific plots	8
CASE 4 specific plots	9
Functions	10

Load Videos

```
clear all; close all; clc

%case 1
load("HW3_vids/cam1_1.mat")
load("HW3_vids/cam2_1.mat")
load("HW3_vids/cam3_1.mat")
%case 2
load("HW3_vids/cam1_2.mat")
load("HW3_vids/cam2_2.mat")
load("HW3_vids/cam3_2.mat")
%case 3
load("HW3_vids/cam1_3.mat")
load("HW3_vids/cam2_3.mat")
load("HW3_vids/cam3_3.mat")
%case 4
load("HW3_vids/cam1_4.mat")
load("HW3_vids/cam2_4.mat")
load("HW3_vids/cam3_4.mat")
```

Identify Initial Tracking Point

```
figure(1)

subplot(4,3,1); show1stframe(vidFrames1_1,321,231); title("Case 1  
Video 1")
subplot(4,3,2); show1stframe(vidFrames2_1,276,274); title("Case 1  
Video 2")
subplot(4,3,3); show1stframe(vidFrames3_1,321,271); title("Case 1  
Video 3")

subplot(4,3,4); show1stframe(vidFrames1_2,322,311); title("Case 2  
Video 1")
```

```

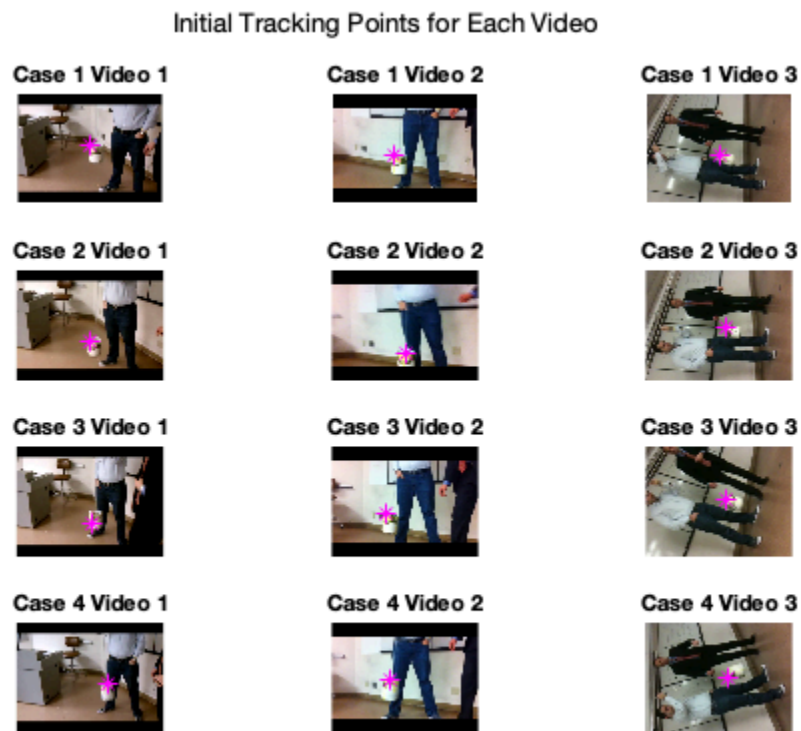
subplot(4,3,5); show1stframe(vidFrames2_2,321,359); title("Case 2
Video 2")
subplot(4,3,6); show1stframe(vidFrames3_2,349,246); title("Case 2
Video 3")

subplot(4,3,7); show1stframe(vidFrames1_3,331,339); title("Case 3
Video 1")
subplot(4,3,8); show1stframe(vidFrames2_3,239,294); title("Case 3
Video 2")
subplot(4,3,9); show1stframe(vidFrames3_3,355,231); title("Case 3
Video 3")

subplot(4,3,10); show1stframe(vidFrames1_4,403,263); title("Case 4
Video 1")
subplot(4,3,11); show1stframe(vidFrames2_4,255,245); title("Case 4
Video 2")
subplot(4,3,12); show1stframe(vidFrames3_4,363,237); title("Case 4
Video 3")

sgtitle("Initial Tracking Points for Each Video")

```



Manual adjustments to optimize point tracking results

```
vidFrames1_2(268, 349, :,237)=255;
```

```
vidFrames2_2(314, 299,:,7)=255;  
vidFrames2_2(301, 301,:,8)=255;  
vidFrames3_2(241, 326,:,236)=255;
```

Create x and y coordinate vectors tracking the top of the paint can through each video

```
%see end of script for explanations of functions  
[x1a,y1a]=track_pt(vidFrames1_1,321,231,20); %case 1 vid 1  
[x1b,y1b]=track_pt(vidFrames2_1,276,274,20); %case 1 vid 2  
[x1c,y1c]=track_pt(vidFrames3_1,321,271,20); %case 1 vid 3  
  
[x2a,y2a]=track_pt(vidFrames1_2,322,311,16); %case 2 vid 1  
[x2b,y2b]=track_pt(vidFrames2_2,316,360,35); %case 2 vid 2  
[x2c,y2c]=track_pt(vidFrames3_2,349,246,31); %case 2 vid 3  
  
[x3a,y3a]=track_pt(vidFrames1_3,331,339,10); %case 3 vid 1  
[x3b,y3b]=track_red_pt(vidFrames2_3,253,293,20); %case 3 vid 2  
[x3c,y3c]=track_pt(vidFrames3_3,355,231,20); %case 1 vid 3  
  
[x4a,y4a]=track_w_red_pt(vidFrames1_4,403,263,250,16); %case 4 vid 1  
[x4b,y4b]=track_w_red_pt(vidFrames2_4,255,245,250,15); %case 4 vid 2  
[x4c,y4c]=track_w_red_pt(vidFrames3_4,363,237,240,15); %case 4 vid 3
```

Play videos with point tracking

```
%Commenting out this section. These videos should all be viewed to  
make  
%sure the vectors created above are tracking the paint can as expected  
  
%{  
figure(2)  
%case 1  
track_vid(vidFrames1_1,x1a,y1a) %plays video 1 with tracked point  
track_vid(vidFrames2_1,x1b,y1b) %plays video 2 with tracked point  
track_vid(vidFrames3_1,x1c,y1c) %plays video 3 with tracked point  
%case 2  
track_vid(vidFrames1_2,x2a,y2a)  
track_vid(vidFrames2_2,x2b,y2b)  
track_vid(vidFrames3_2,x2c,y2c)  
%case3  
track_vid(vidFrames1_3,x3a,y3a)  
track_vid(vidFrames2_3,x3b,y3b)  
track_vid(vidFrames3_3,x3c,y3c)  
%case 4  
track_vid(vidFrames1_4,x4a,y4a)  
track_vid(vidFrames2_4,x4b,y4b)  
track_vid(vidFrames3_4,x4c,y4c)  
%}
```

Subtract off the mean, align vectors, and truncate x and y coordinate vectors

```
tp1=200; tp2=297; tp3=226; tp4=385;
%case1
x1a=x1a(1:tp1)-mean(x1a(1:tp1));
x1b=x1b(1+9:tp1+9)-mean(x1b(1+9:tp1+9)); %altered to align peaks and
troughs
x1c=x1c(1:tp1)-mean(x1c(1:tp1));
y1a=y1a(1:tp1)-mean(y1a(1:tp1));
y1b=y1b(1+9:tp1+9)-mean(y1b(1+9:tp1+9)); %altered to align peaks and
troughs
y1c=y1c(1:tp1)-mean(y1c(1:tp1));
%case2
x2a=x2a(1:tp2)-mean(x2a(1:tp2));
x2b=x2b(1+26:tp2+26)-mean(x2b(1+26:tp2+26));
x2c=x2c(1+10:tp2+10)-mean(x2c(1+10:tp2+10));
y2a=y2a(1:tp2)-mean(y2a(1:tp2));
y2b=y2b(1+26:tp2+26)-mean(y2b(1+26:tp2+26));
y2c=y2c(1+10:tp2+10)-mean(y2c(1+10:tp2+10));
%case3
x3a=x3a(1+11:tp3+11)-mean(x3a(1+11:tp3+11));
x3b=x3b(1:tp3)-mean(x3b(1:tp3));
x3c=x3c(1+4:tp3+4)-mean(x3c(1+4:tp3+4));
y3a=y3a(1+11:tp3+11)-mean(y3a(1+11:tp3+11));
y3b=y3b(1:tp3)-mean(y3b(1:tp3));
y3c=y3c(1+4:tp3+4)-mean(y3c(1+4:tp3+4));
%case4
x4a=x4a(1:tp4)-mean(x4a(1:tp4));
x4b=x4b(1+18:tp4+18)-mean(x4b(1+18:tp4+18));
x4c=x4c(1+7:tp4+7)-mean(x4c(1+7:tp4+7));
y4a=y4a(1:tp4)-mean(y4a(1:tp4));
y4b=y4b(1+18:tp4+18)-mean(y4b(1+18:tp4+18));
y4c=y4c(1+7:tp4+7)-mean(y4c(1+7:tp4+7));
```

Oh the SVD won't let me be

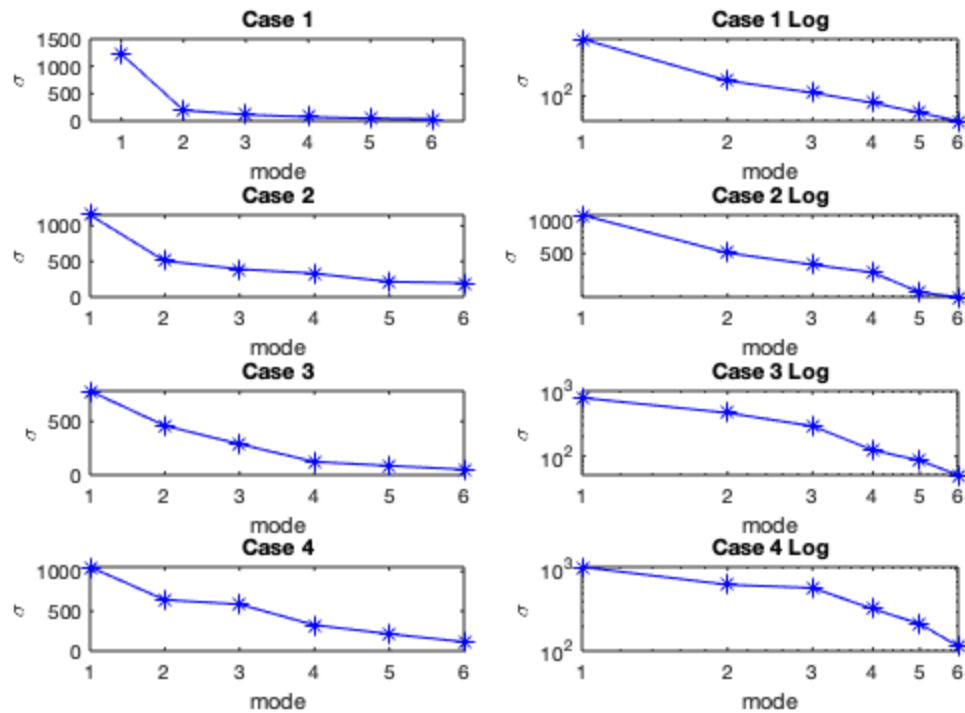
```
%CASE 1
A1=[x1a; y1a; x1b; y1b; x1c; y1c]';
[U1,S1,V1]=svd(A1,0);
sig1=diag(S1);%Create vector of singular values
%CASE 2
A2=[x2a; y2a; x2b; y2b; x2c; y2c]';
[U2,S2,V2]=svd(A2,0);
sig2=diag(S2);
%CASE 3
A3=[x3a; y3a; x3b; y3b; x3c; y3c]';
[U3,S3,V3]=svd(A3,0);
sig3=diag(S3);
%CASE 4
A4=[x4a; y4a; x4b; y4b; x4c; y4c]';
```

```
[U4,S4,V4]=svd(A4,0);  
sig4=diag(S4);
```

Plot singular values to identify principal components

```
figure(2)  
subplot(4,2,1); plot(sig1,"b*-"); axis([.5 6.5 -.5  
1500]);xlabel('mode');ylabel('\sigma'); title("Case 1")  
subplot(4,2,2); loglog(sig1,"b*-");  
xlabel('mode');ylabel('\sigma');title("Case 1 Log")  
  
subplot(4,2,3); plot(sig2,"*-",'Color','b') ;  
xlabel('mode');ylabel('\sigma'); title("Case 2")  
subplot(4,2,4); loglog(sig2,"*-",'Color','b') ;  
xlabel('mode');ylabel('\sigma'); title("Case 2 Log")  
  
subplot(4,2,5); plot(sig3,"*-",'Color','b') ;  
xlabel('mode');ylabel('\sigma'); title("Case 3")  
subplot(4,2,6); loglog(sig3,"*-",'Color','b') ;  
xlabel('mode');ylabel('\sigma'); title("Case 3 Log")  
  
subplot(4,2,7);  
plot(sig4,"*-",'Color','b') ;xlabel('mode');ylabel('\sigma');  
title("Case 4")  
subplot(4,2,8); loglog(sig4,"*-",'Color','b') ;  
xlabel('mode');ylabel('\sigma');title("Case 4 Log")  
  
sgtitle("Plots of Singular Values")
```

Plots of Singular Values



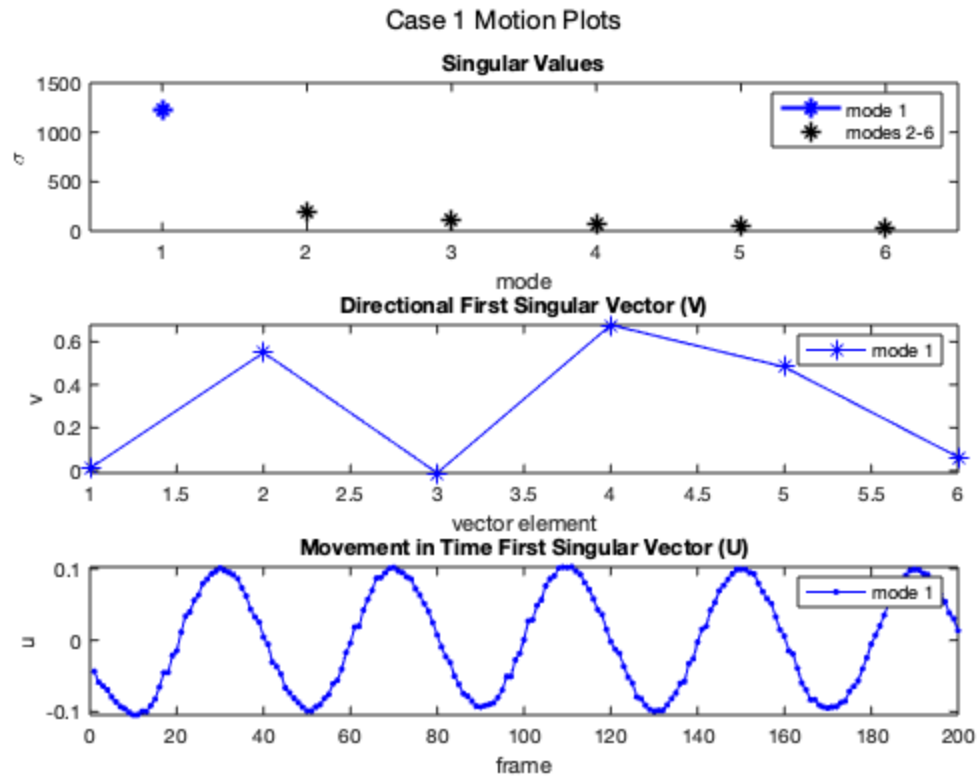
CASE 1 specific plots

```
figure(3)

subplot(3,1,1);
plot(1,sig1(1), "b*-", 'Linewidth',2), hold on ;
plot([2:6],sig1(2:6), "k*", 'Linewidth',1.5);
xlabel('mode');ylabel('\sigma');
title("Singular Values")
legend("mode 1", "modes 2-6")
axis([.5 6.5 -.5 1500])

subplot(3,1,2); plot(V1(:,1), "b*-"); xlabel('vector
element');ylabel('v');title("Directional First Singular Vector (V)")
legend("mode 1")

subplot(3,1,3); plot(U1(:,1), "b.-"); xlabel('frame');ylabel('u');
title("Movement in Time First Singular Vector (U)")
legend("mode 1")
sgtitle("Case 1 Motion Plots")
```



CASE 2 specific plots

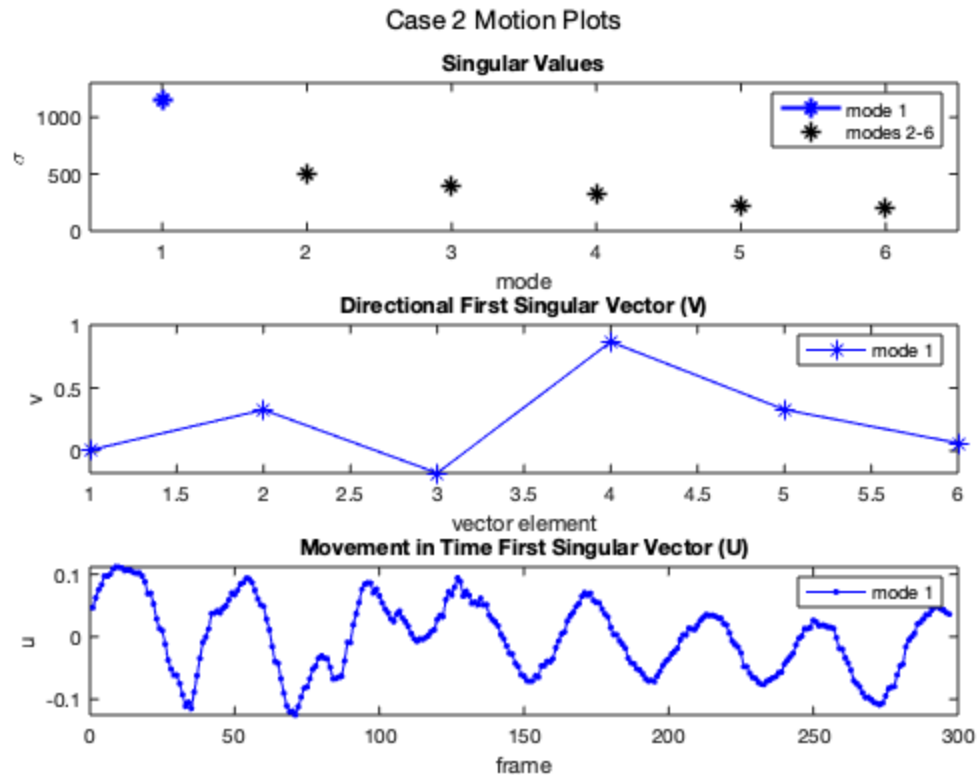
```
figure(4)

subplot(3,1,1);
plot(1,sig2(1), "b*-", 'Linewidth',2), hold on ;
plot([2:6],sig2(2:6), "k*", 'Linewidth',1.5);
xlabel('mode');ylabel('\sigma');
title("Singular Values")
legend("mode 1", "modes 2-6")
axis([.5 6.5 -.5 1300])

subplot(3,1,2); plot(V2(:,1), "b*-");
xlabel('vector element'); ylabel('v'); title("Directional First
Singular Vector (V)");legend("mode 1")

subplot(3,1,3); plot(U2(:,1), "b.-");
xlabel('frame'); ylabel('u'); title("Movement in Time First Singular
Vector (U)");legend("mode 1")

sgtitle("Case 2 Motion Plots")
```



CASE 3 specific plots

```
figure(5)
subplot(3,1,1)
plot(1,sig3(1), '*', 'Color', '#0097A1', 'Linewidth', 2), hold on
plot(2,sig3(2), '*', 'Color', '#B13CD6', 'Linewidth', 2)
plot(3,sig3(3), '*', 'Color', '#F57D19', 'Linewidth', 2)
plot([4:6],sig3(4:6), '*', 'Color', '#928E9A', 'Linewidth', 2)
legend("mode 1", "mode 2", "mode 3", "modes 4-6")
xlabel('mode'); ylabel('\sigma');
title("Singular Values")

subplot(3,1,2)
plot(V3(:,1), '-.', 'Color', '#0097A1', 'Linewidth', 1.5), hold on,
plot(V3(:,2), '-.', 'Color', '#B13CD6', 'Linewidth', 1.5),
plot(V3(:,3), '-.', 'Color', '#F57D19', 'Linewidth', 1.5)
title("Direction (V singular vectors)");
xlabel('vector element'); ylabel('v');
legend("mode 1", "mode 2", "mode 3")

subplot(3,1,3)
plot(U3(:,1), '-.', 'Color', '#0097A1', 'Linewidth', 1), hold on,
plot(U3(:,2), '-.', 'Color', '#B13CD6', 'Linewidth', 1),
plot(U3(:,3), '-.', 'Color', '#F57D19', 'Linewidth', 1)
title("Motion in Time (U singular vectors)");
```

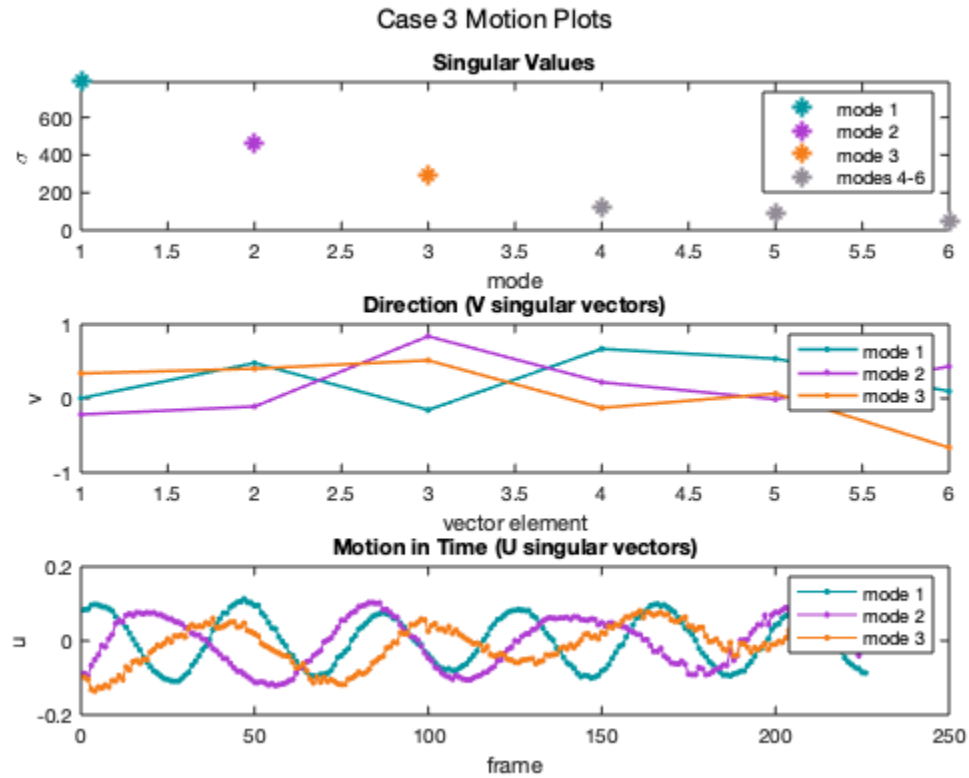


```

xlabel('frame'); ylabel('u');
legend("mode 1","mode 2","mode 3")

sgtitle("Case 3 Motion Plots")

```



CASE 4 specific plots

```

figure(6)
subplot(3,1,1)
plot(1,sig4(1),'*','Color','#0097A1','Linewidth',2), hold on
plot(2,sig4(2),'*','Color','#B13CD6','Linewidth',2)
plot(3,sig4(3),'*','Color','#F57D19','Linewidth',2)
plot([4:6],sig4(4:6),'*','Color','#928E9A','Linewidth',2)
xlabel('mode');ylabel('\sigma');
legend("mode 1","mode 2","mode 3","modes 4-6")
title("Singular Values")

subplot(3,1,2)
plot(V4(:,1),'.-','Color','#0097A1','Linewidth',1.5), hold on,
plot(V4(:,2),'.-','Color','#B13CD6','Linewidth',1.5),
plot(V4(:,3),'.-','Color','#F57D19','Linewidth',1.5)
xlabel('vector element'); ylabel('v');
title("Direction (V singular vectors)")
legend("mode 1","mode 2","mode 3")

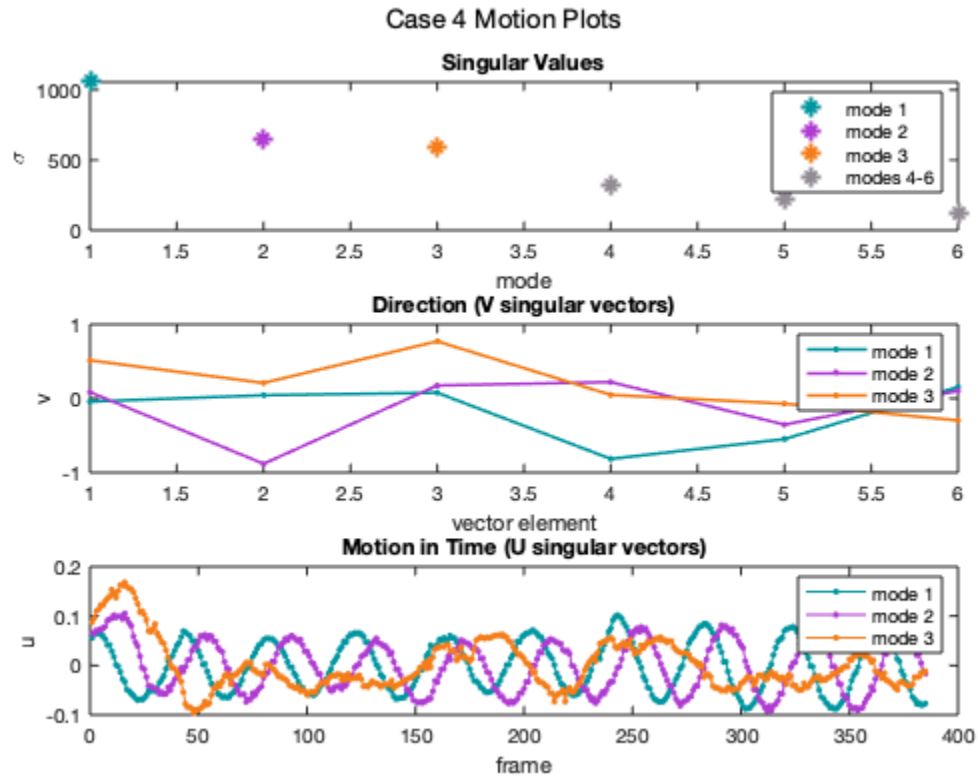
```

```

subplot(3,1,3)
plot(U4(:,1),'.-','Color','#0097A1','Linewidth',1), hold on,
plot(U4(:,2),'.-','Color','#B13CD6','Linewidth',1),
plot(U4(:,3),'.-','Color','#F57D19','Linewidth',1)
xlabel('frame'); ylabel('u');
title("Motion in Time (U singular vectors)")
legend("mode 1","mode 2","mode 3")

sgtitle("Case 4 Motion Plots")

```



Functions

```

%{
Point Tracking Function Descriptions:
-----
track_pt(vidnm,x_l,y_l,w) - finds the pixel with the maximum color sum
    in
        the window of width and height 2w centers around the
    previous frame's
        tracked point
track_red_pt -- Like track_pt but looks for most red point by
    maximizing
        the first RGB value where the other two values are below
    175.

```

```

track_w_red_pt -- looks for brightest white where all RGB values are
above
    a given threshold. If no pixel is found matching this
criteria, it chooses
    the reddest pixel
%}

function [x,y] = track_pt(vidnm, x_1, y_1, winsz)
    x_w=x_1; y_w=y_1; %center of window coordinates
    x=x_1; y=y_1; %vectors to store of x y coordinates
    for i=2:size(vidnm,4)
        colsum=0;
        frame=vidnm( :, :, :, i);
        for jx = x_w-winsz:x_w+winsz
            for jy = y_w-winsz:y_w+winsz
                col=frame(jy,jx,:); %get color of pixel
                newcolsum =
double(col(1))+double(col(2))+double(col(3));
                if newcolsum>colsum
                    colsum = newcolsum;
                    x_max=jx;
                    y_max=jy;
                end
            end
        end
        x=[x,x_max];
        y=[y,y_max];
        x_w = x_max;
        y_w = y_max;
    end
end

function [x,y] = track_w_red_pt(vidnm, x_1, y_1, wthresh, winsz)
    x_w=x_1; y_w=y_1; %center of window coordinates
    x=x_1; y=y_1; %vectors to store of x y coordinates
    x_maxr=x_1; y_maxr=y_1;
    for i=2:size(vidnm,4)
        colr=0;
        colsum=0;
        frame=vidnm( :, :, :, i);
        for jx = x_w-winsz:x_w+winsz
            for jy = y_w-winsz:y_w+winsz
                col=frame(jy,jx,:); %get color of pixel
                newcolr = double(col(1));
                newcolsum =
double(col(1))+double(col(2))+double(col(3));
                if newcolsum>colsum && double(col(1))>wthresh &&
double(col(2))>wthresh && double(col(3))>wthresh
                    colr = newcolr;
                    colsum=newcolsum;
                    x_maxr=jx;
                    y_maxr=jy;
                elseif newcolr>=colr && double(col(2))<175 &&
double(col(3))<175

```

```

        colr = newcolr;
        colsum=newcolsum;
        x_maxr=jx;
        y_maxr=jy;
    end
end
end
x=[x,x_maxr];
y=[y,y_maxr];
x_w = x_maxr;
y_w = y_maxr;
end
end

function [x,y] = track_red_pt(vidnm, x_1, y_1,winsz)
    x_w=x_1; y_w=y_1; %center of window coordinates
    x=x_1; y=y_1; %vectors to store of x y coordinates
    x_maxr=x_1; y_maxr=y_1;
    for i=2:size(vidnm,4)
        colr=0;
        frame=vidnm( :, :, :,i);
        for jx = x_w-winsz:x_w+winsz
            for jy = y_w-winsz:y_w+winsz
                col=frame(jy,jx,:);%get color of pixel
                newcolr = double(col(1));
                if newcolr>=colr && double(col(2))<175 &&
double(col(3))<175
                    colr = newcolr;
                    x_maxr=jx;
                    y_maxr=jy;
                end
            end
        end
        x=[x,x_maxr];
        y=[y,y_maxr];
        x_w = x_maxr;
        y_w = y_maxr;
    end
end

%Plays video with point tracker at x y coordinates given in xvec and
yvec
function track_vid(vidnm, xvec, yvec )
    for i=1:size(vidnm,4)
        figure(1)
        imshow(uint8(vidnm( :, :, :,i))); hold on
        plot(xvec(i),yvec(i),"m*","linewidth",[1.5])
        hold off
    end
end

%Shows first frame of the video with highlighted initial point
function show1stframe(vidnm, x_1,y_1)

```

```
    imshow(uint8(vidnm( :, :, :,1))); hold on
    plot(x_1,y_1,"m*")
end
```

Published with MATLAB® R2019b