01

March 28, 2018

1 Assignment 1

1.1 Exercise 1.1

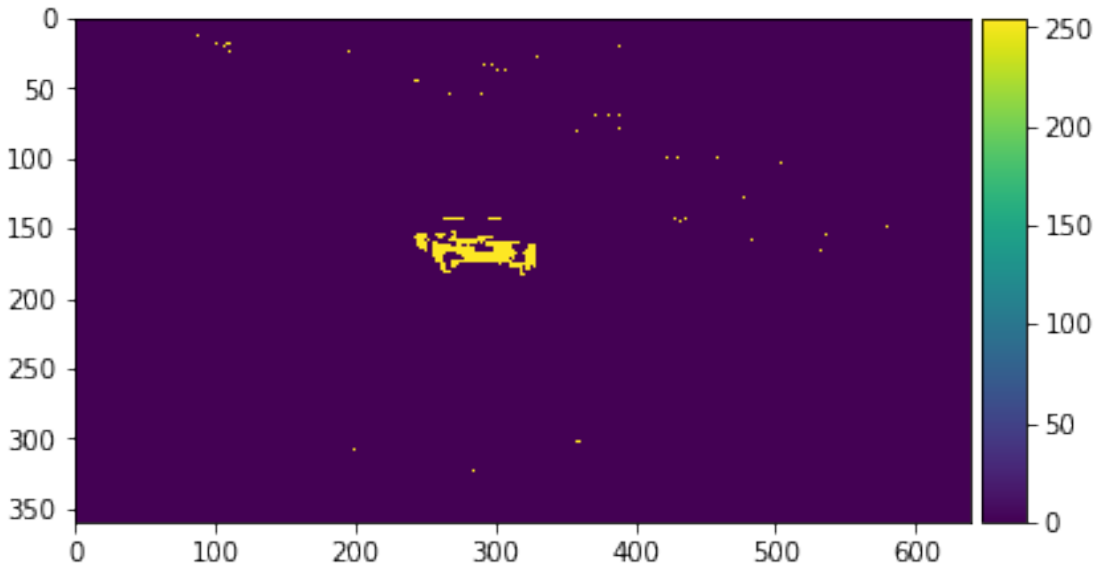
rough sketch for ex 1.2:

1. define the positive“ subspace P in the RGB cube
2. iterate over all pixels in I and check if in P or ~P
3. write result to new image
4. play around with size and shape of P and display binary image (**RESULT**)

```
In [14]: %matplotlib inline

         from skimage import io, data, color
         from skimage.morphology import erosion, disk, dilation, square
         import numpy as np
         import matplotlib.pyplot as plt
         import warnings; warnings.filterwarnings('ignore')

         image = io.imread('images/racecar/010.jpeg')
         height, width, _ = image.shape
         print(width, height)

         def create_mask(image, width, height):
             mask = np.zeros_like(image[:,:,0])

             for y in range(0,height):
                 for x in range(0,width):
                     rgb = [image[y,x,0], image[y,x,1], image[y,x,2]]
                     rgb.sort()
                     color_min = rgb[1] - 40
                     color_max = rgb[1] + 40

                     # filter greyish color
                     if (color_min < rgb[0] < color_max and color_min < rgb[2] < color_max):
                         continue

                     # get blue color
```

```
                if image[y,x,2] > 110 and image[y,x,0:1] < 220 :
                    mask[y,x] = 255

        return mask

    mask = create_mask(image, width, height)

    io.imshow(mask)
    io.show()
    # RESULT
```

```
640 360
```



1.2 Exercise 1.2

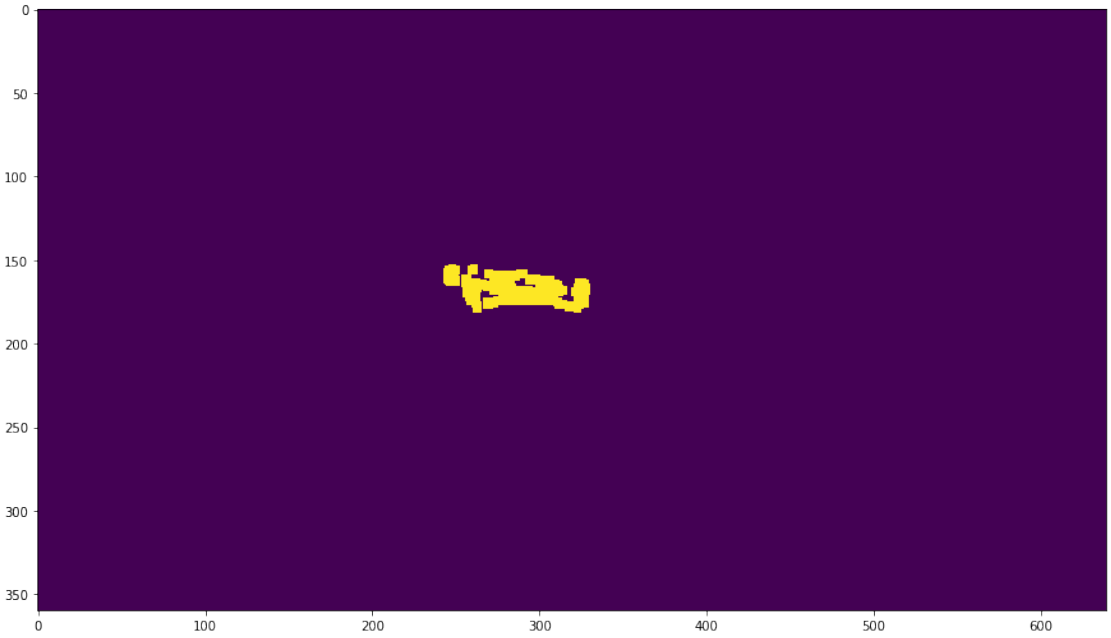
- starting from the binary color detection image
- erase noise with an erosion operation
- dilate once to get original size of object
- find connected components with one-pass algorithm
- extract bounding box on the fly
- draw bounding box on original image (**RESULT**)

```
In [4]: from skimage.draw import polygon_perimeter

    selem = square(3)
    eroded = erosion(mask, selem)
    dilated = dilation(eroded, selem)
```

```
dilated = dilation(dilated, selem)

mask = dilated
height, width = mask.shape

#test3color = io.imread('images/test3.png')
#test3 = color.rgb2gray(test3color)
#height, width = test3.shape

class BBox:
    # static var - can be accessed by BBox.bboxes - clear this before use again
    bboxes = dict()     # todo: rename

    minX = 9999
    minY = 9999

    maxX = 0
    maxY = 0

    def add(self, x, y):
        BBox.bboxes[(x,y)] = self
        if self.minX > x:
            self.minX = x
        if self.maxX < x:
            self.maxX = x
        if self.minY > y:
            self.minY = y
        if self.maxY < y:
            self.maxY = y

    def join(self, other_bbox):
        for k,v in BBox.bboxes.items():
            if v == other_bbox:
                self.add(k[0], k[1])

    def get_size(self):
        width = self.maxX - self.minX
        height = self.maxY - self.minY
        return width * height

    @staticmethod
    def add_create_bbox(x, y):
        if (x-1,y) in BBox.bboxes and not (x,y-1) in BBox.bboxes:
            # add
            BBox.bboxes[(x-1,y)].add(x,y)
        elif (x,y-1) in BBox.bboxes and not (x-1,y) in BBox.bboxes:
            # add
            BBox.bboxes[(x,y-1)].add(x,y)
```

```
            elif (x,y-1) in BBox.bboxes and (x-1,y) in BBox.bboxes:
                # join
                BBox.bboxes[(x-1,y)].add(x,y)
                BBox.bboxes[(x-1,y)].join(BBox.bboxes[(x,y-1)])
            else:
                # create new
                box = BBox()
                BBox.bboxes[(x,y)] = box


    def compute_bounding_boxes(mask, img, draw_boxes = True):
        BBox.bboxes.clear()
        height, width = mask.shape

        # calculate bboxes
        for y in range(0,height):
            for x in range (0,width):
                if mask[y, x] == 255:  # use 1.0 for test3
                    BBox.add_create_bbox(x, y)

        # draw bboxes
        if draw_boxes:
            for box in set(BBox.bboxes.values()):
                rr, cc = polygon_perimeter([box.minY, box.maxY, box.maxY, box.maxY, box.ma
                                            [box.minX, box.minX, box.minX, box.maxX, box.maxX,
                                            img.shape)
                img[rr, cc] = [255,255,255] # use [0,0,255,255] for test 3

        return set(BBox.bboxes.values())


    bboxes = compute_bounding_boxes(mask, image)
    print(len(bboxes))

    plt.figure(1, figsize=(15, 20))
    plt.imshow(mask)
    plt.figure(2, figsize=(15, 20))
    plt.imshow(image)
    plt.show()
```

2



1.2.1 Number of component

```
In [5]: from skimage import measure
        import numpy as np
```

```
print(len(bboxes))
```

```
2
```

1.2.2 Bounding Box

```
In [6]: from skimage.measure import regionprops

        import math
        import cv2

        # draw bboxes
        for box in set(BBox.bboxes.values()):
            rr, cc = polygon_perimeter([box.minY, box.maxY, box.maxY, box.maxY, box.maxY, box.
                                       [box.minX, box.minX, box.minX, box.maxX, box.maxX, box.maxX
                                       image.shape)

            image[rr, cc] = [255,255,0] # use [0,0,255,255] for test 3

        plt.figure(figsize=(15, 20))
        plt.imshow(image)
        plt.show()
```

1.3 Exercise 1.3

- use your color detection and connected components algorithm

- implement simplest tracking algorithm
- draw history of all previous points on frame (**RESULT**)

```
In [8]: def get_file_name(i):
            if i < 10:
                return "images/racecar/00" + str(i) + ".jpeg"
            if i < 100:
                return "images/racecar/0" + str(i) + ".jpeg"
            return "images/racecar/" + str(i) + ".jpeg"

        def draw_line(img, coord_list):
            coords = list(map(list, zip(*coord_list)))
            rr, cc = polygon_perimeter(coords[0], coords[1], img.shape, clip=False)
            img[cc,rr] = [0,0,255] # use [0,0,255,255] for test 3
            return img

In [11]: tracking_points = []

         for i in range(0 ,150, 1):
             print(i)
             tmp_image = io.imread(get_file_name(i))
             height, width, _ = tmp_image.shape
             tmp_mask = create_mask(tmp_image, width, height)

             eroded = erosion(tmp_mask, selem)
             dilated = dilation(eroded, selem)

             tmp_box = compute_bounding_boxes(dilated, tmp_image, draw_boxes = False)
             print("Ammount of bboxes: " + str(len(tmp_box)))

             l = list(tmp_box)

             # use greatest bounding box
             largest = None
             size = 0
             for b in l:
                 if b.get_size() > size:
                     largest = b
                     size = b.get_size()

             if largest:
                 tracking_points.append( (largest.minX, largest.minY) )
                 rr, cc = polygon_perimeter([largest.minY, largest.maxY, largest.maxY, largest
                                             largest.maxY, largest.minY, largest.minY],
                                            [largest.minX, largest.minX, largest.minX, largest
                                             largest.maxX, largest.maxX, largest.minX],
                                             tmp_image.shape)
                 tmp_image[rr, cc] = [255,255,255] # use [0,0,255,255] for test 3
```

```
            if (len(tracking_points) > 2):
                tmp_image = draw_line(tmp_image, tracking_points)

            plt.figure(i, figsize=(15,20))
            plt.imshow(tmp_image)


        # plt.figure(8, figsize=(15,20))
        # tmp_image = io.imread(get_file_name(151))
        # tmp_image = draw_line(tmp_image, tracking_points)
        plt.imshow(tmp_image)

        #plt.figure(figsize=(15,20))
        plt.show()
```

```
0
Ammount of bboxes: 5
1
Ammount of bboxes: 5
2
Ammount of bboxes: 3
3
Ammount of bboxes: 3
4
Ammount of bboxes: 4
5
Ammount of bboxes: 5
6
Ammount of bboxes: 6
7
Ammount of bboxes: 4
8
Ammount of bboxes: 4
9
Ammount of bboxes: 6
10
Ammount of bboxes: 4
11
Ammount of bboxes: 6
12
Ammount of bboxes: 5
13
Ammount of bboxes: 7
14
Ammount of bboxes: 2
15
Ammount of bboxes: 3
```

```
16
Ammount of bboxes: 5
17
Ammount of bboxes: 6
18
Ammount of bboxes: 6
19
Ammount of bboxes: 4
20
Ammount of bboxes: 10
21
Ammount of bboxes: 10
22
Ammount of bboxes: 7
23
Ammount of bboxes: 6
24
Ammount of bboxes: 9
25
Ammount of bboxes: 6
26
Ammount of bboxes: 7
27
Ammount of bboxes: 7
28
Ammount of bboxes: 8
29
Ammount of bboxes: 6
30
Ammount of bboxes: 8
31
Ammount of bboxes: 8
32
Ammount of bboxes: 10
33
Ammount of bboxes: 10
34
Ammount of bboxes: 21
35
Ammount of bboxes: 15
36
Ammount of bboxes: 20
37
Ammount of bboxes: 26
38
Ammount of bboxes: 28
39
Ammount of bboxes: 22
```

```
40
Ammount of bboxes: 23
41
Ammount of bboxes: 23
42
Ammount of bboxes: 20
43
Ammount of bboxes: 24
44
Ammount of bboxes: 26
45
Ammount of bboxes: 24
46
Ammount of bboxes: 20
47
Ammount of bboxes: 26
48
Ammount of bboxes: 27
49
Ammount of bboxes: 24
50
Ammount of bboxes: 29
51
Ammount of bboxes: 27
52
Ammount of bboxes: 33
53
Ammount of bboxes: 36
54
Ammount of bboxes: 33
55
Ammount of bboxes: 36
56
Ammount of bboxes: 35
57
Ammount of bboxes: 43
58
Ammount of bboxes: 49
59
Ammount of bboxes: 36
60
Ammount of bboxes: 47
61
Ammount of bboxes: 30
62
Ammount of bboxes: 28
63
Ammount of bboxes: 18
```

```
64
Ammount of bboxes: 27
65
Ammount of bboxes: 20
66
Ammount of bboxes: 25
67
Ammount of bboxes: 26
68
Ammount of bboxes: 21
69
Ammount of bboxes: 17
70
Ammount of bboxes: 25
71
Ammount of bboxes: 24
72
Ammount of bboxes: 27
73
Ammount of bboxes: 29
74
Ammount of bboxes: 35
75
Ammount of bboxes: 22
76
Ammount of bboxes: 27
77
Ammount of bboxes: 24
78
Ammount of bboxes: 27
79
Ammount of bboxes: 24
80
Ammount of bboxes: 33
81
Ammount of bboxes: 40
82
Ammount of bboxes: 39
83
Ammount of bboxes: 43
84
Ammount of bboxes: 50
85
Ammount of bboxes: 55
86
Ammount of bboxes: 49
87
Ammount of bboxes: 59
```

```
88
Ammount of bboxes: 14
89
Ammount of bboxes: 21
90
Ammount of bboxes: 41
91
Ammount of bboxes: 36
92
Ammount of bboxes: 38
93
Ammount of bboxes: 30
94
Ammount of bboxes: 38
95
Ammount of bboxes: 50
96
Ammount of bboxes: 48
97
Ammount of bboxes: 57
98
Ammount of bboxes: 75
99
Ammount of bboxes: 65
100
Ammount of bboxes: 69
101
Ammount of bboxes: 53
102
Ammount of bboxes: 73
103
Ammount of bboxes: 73
104
Ammount of bboxes: 76
105
Ammount of bboxes: 60
106
Ammount of bboxes: 50
107
Ammount of bboxes: 42
108
Ammount of bboxes: 55
109
Ammount of bboxes: 50
110
Ammount of bboxes: 73
111
Ammount of bboxes: 56
```

```
112
Ammount of bboxes: 52
113
Ammount of bboxes: 53
114
Ammount of bboxes: 54
115
Ammount of bboxes: 50
116
Ammount of bboxes: 51
117
Ammount of bboxes: 38
118
Ammount of bboxes: 41
119
Ammount of bboxes: 41
120
Ammount of bboxes: 49
121
Ammount of bboxes: 48
122
Ammount of bboxes: 35
123
Ammount of bboxes: 33
124
Ammount of bboxes: 31
125
Ammount of bboxes: 32
126
Ammount of bboxes: 30
127
Ammount of bboxes: 31
128
Ammount of bboxes: 33
129
Ammount of bboxes: 34
130
Ammount of bboxes: 28
131
Ammount of bboxes: 32
132
Ammount of bboxes: 34
133
Ammount of bboxes: 41
134
Ammount of bboxes: 41
135
Ammount of bboxes: 49
```

```
136
Ammount of bboxes: 42
137
Ammount of bboxes: 54
138
Ammount of bboxes: 57
139
Ammount of bboxes: 58
140
Ammount of bboxes: 61
141
Ammount of bboxes: 55
142
Ammount of bboxes: 55
143
Ammount of bboxes: 57
144
Ammount of bboxes: 58
145
Ammount of bboxes: 47
146
Ammount of bboxes: 50
147
Ammount of bboxes: 50
148
Ammount of bboxes: 46
149
Ammount of bboxes: 43
```


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM.clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

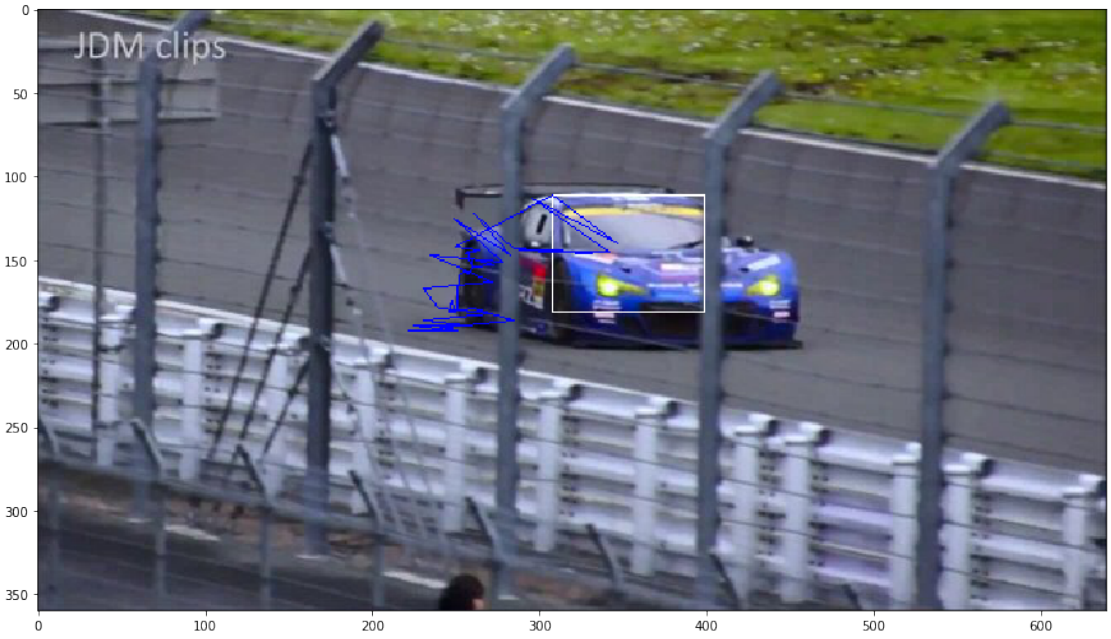

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

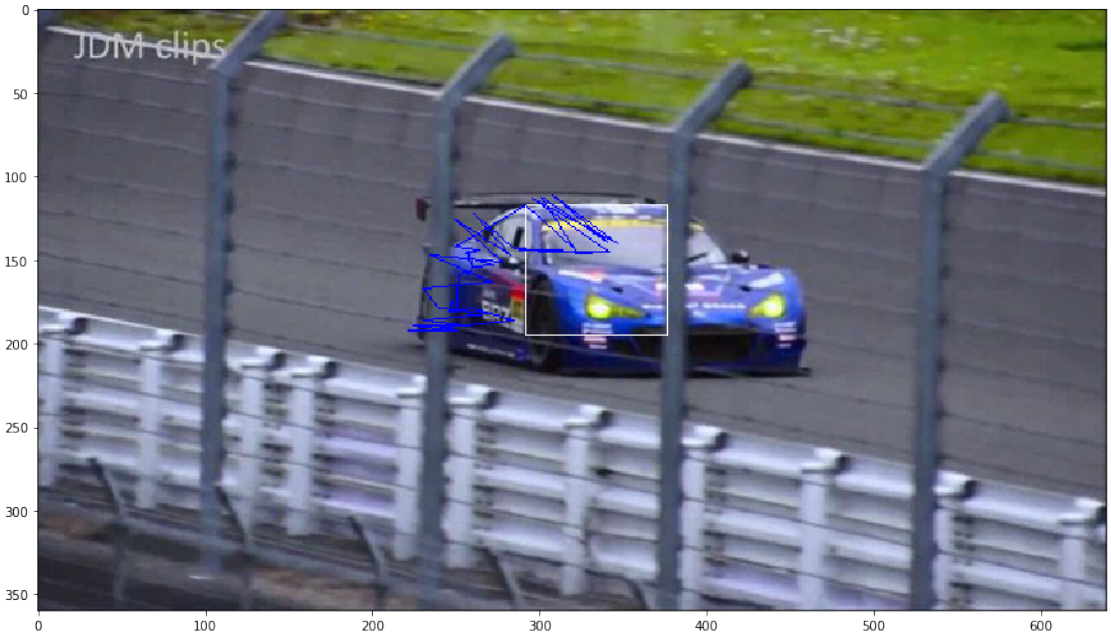

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

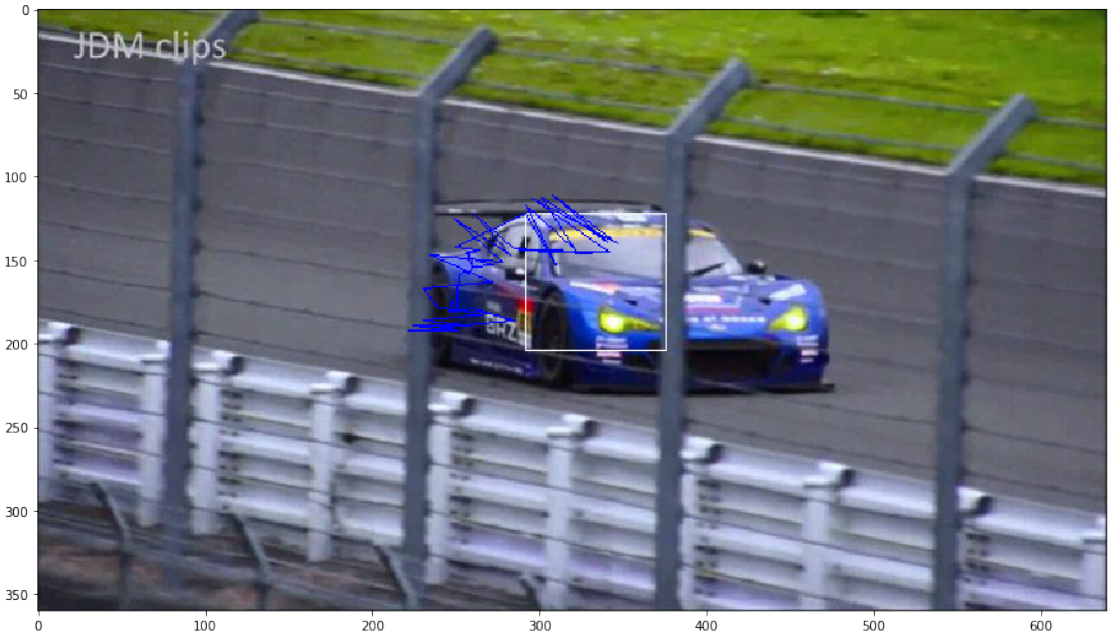

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

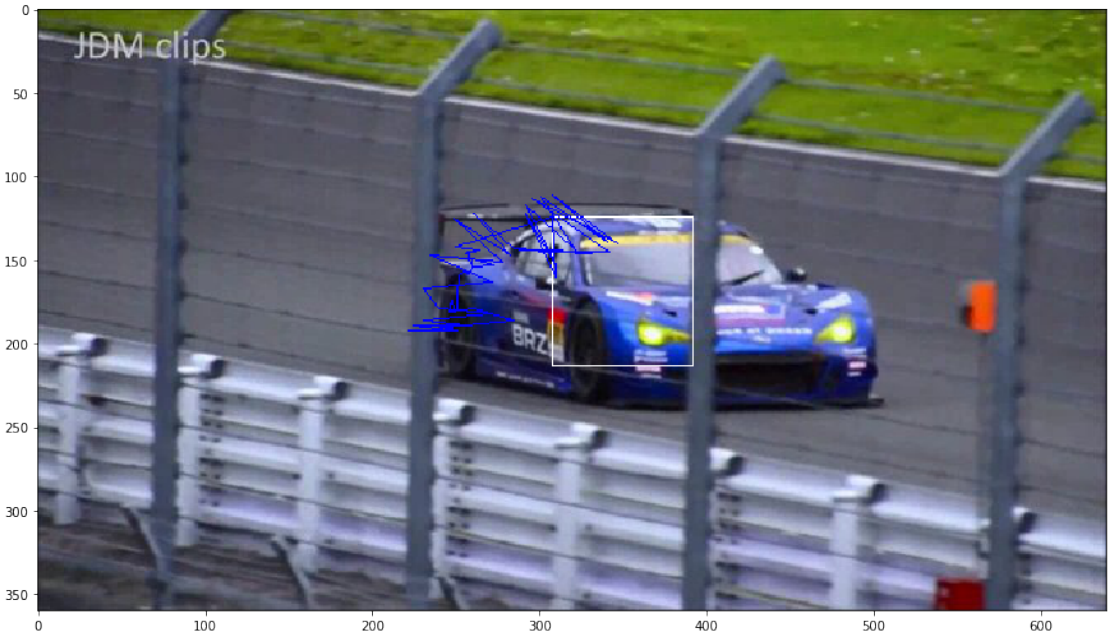

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

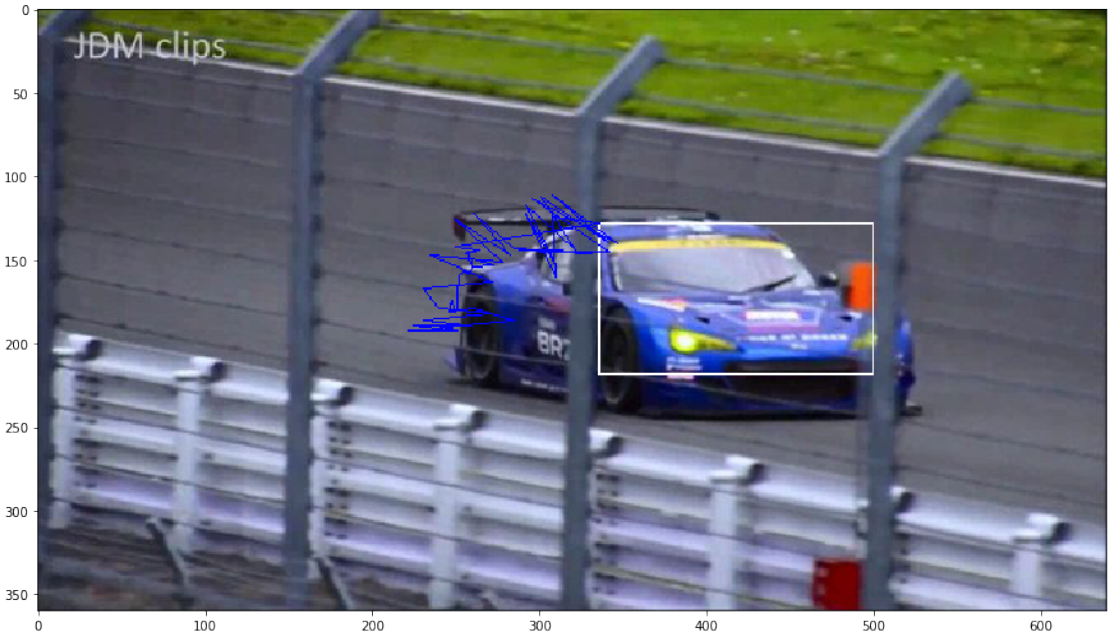

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM-clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

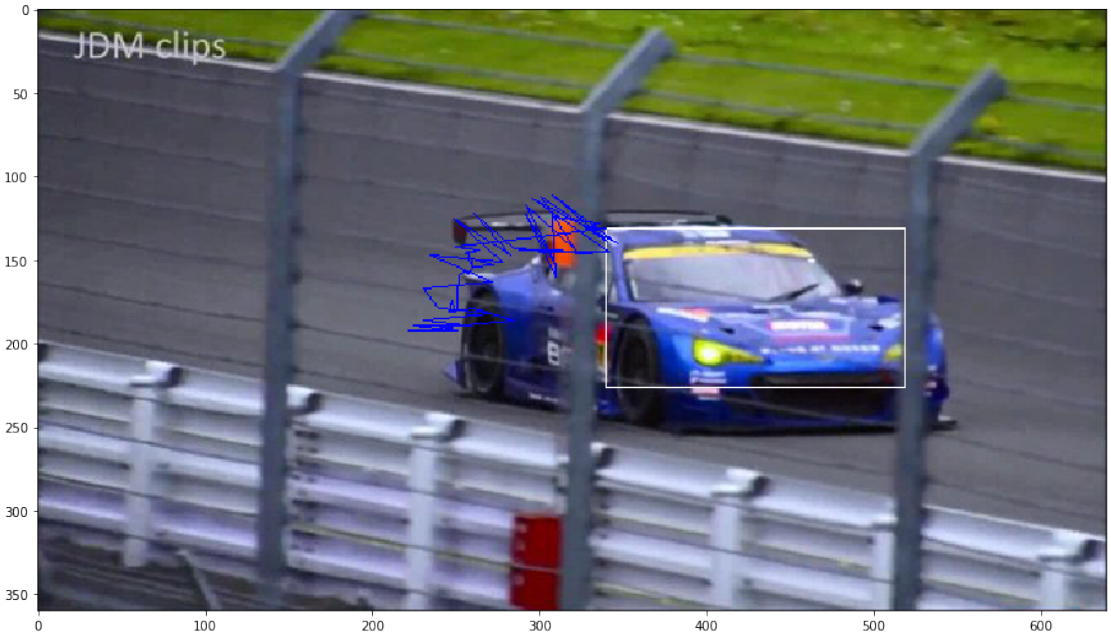

JDM-clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

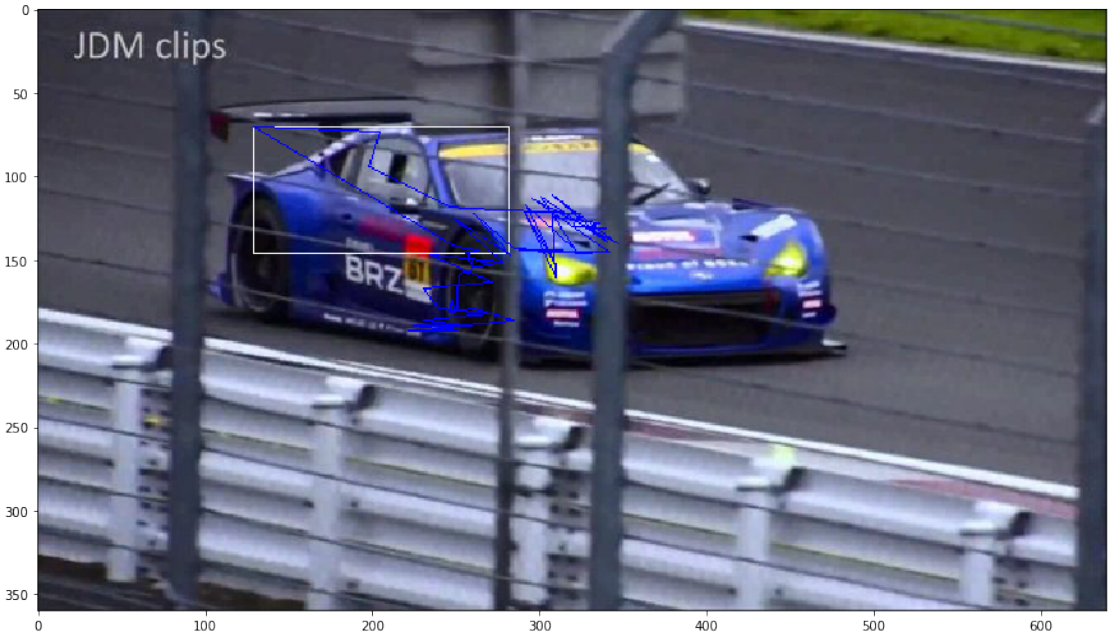

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

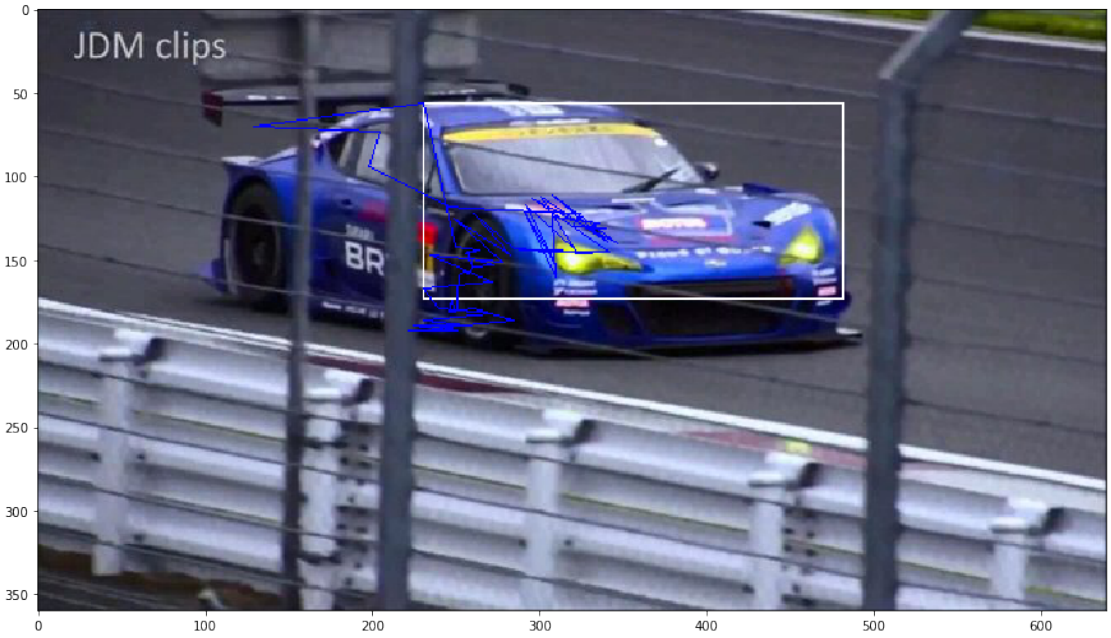

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

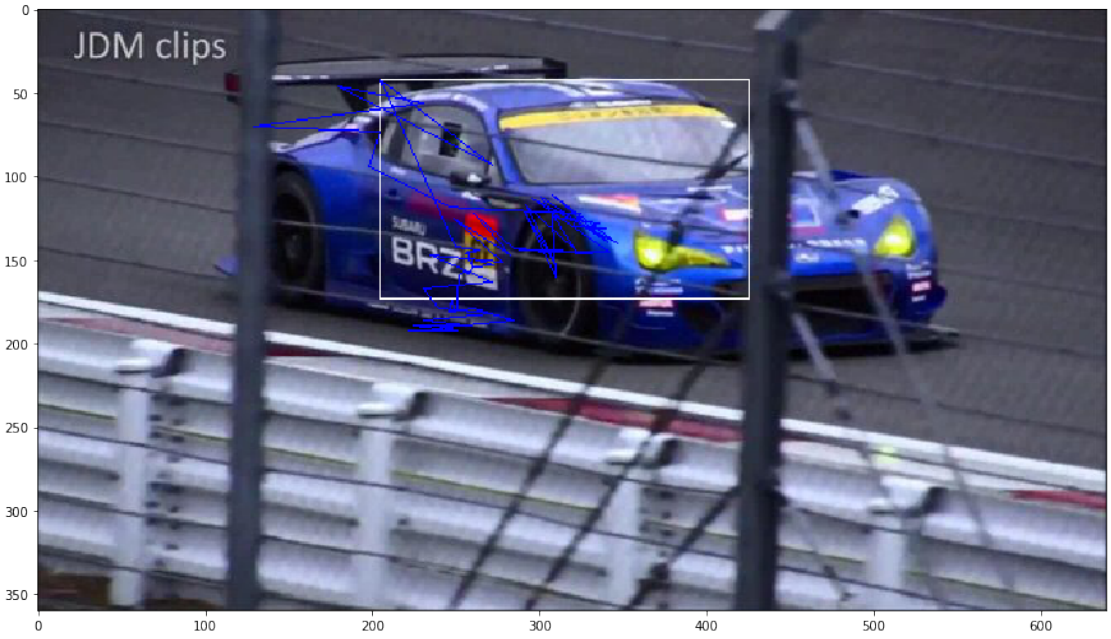

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

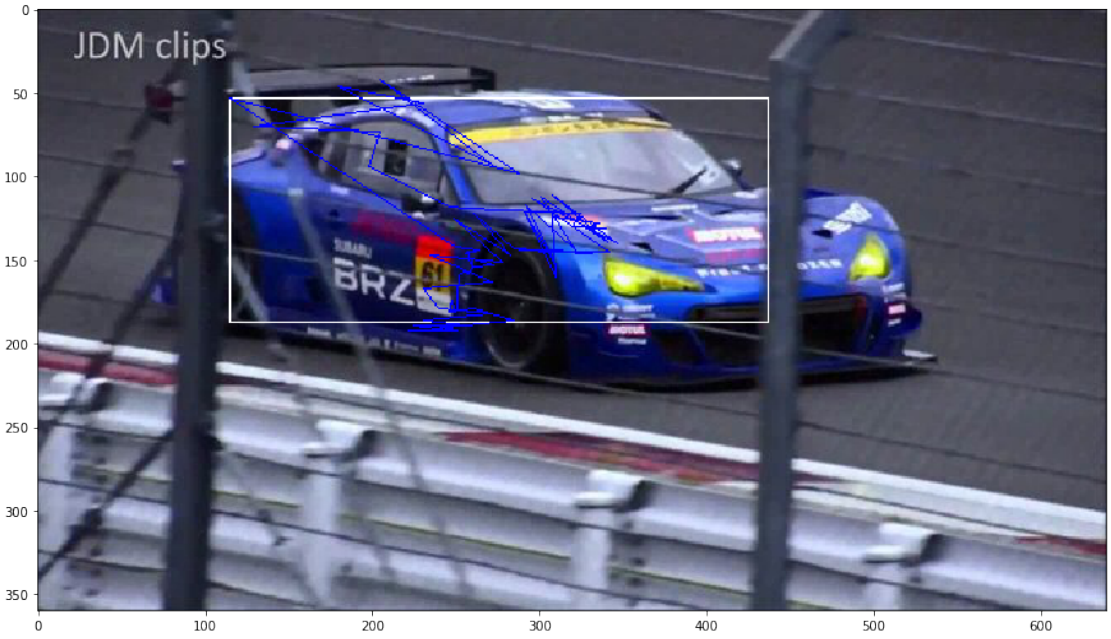

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

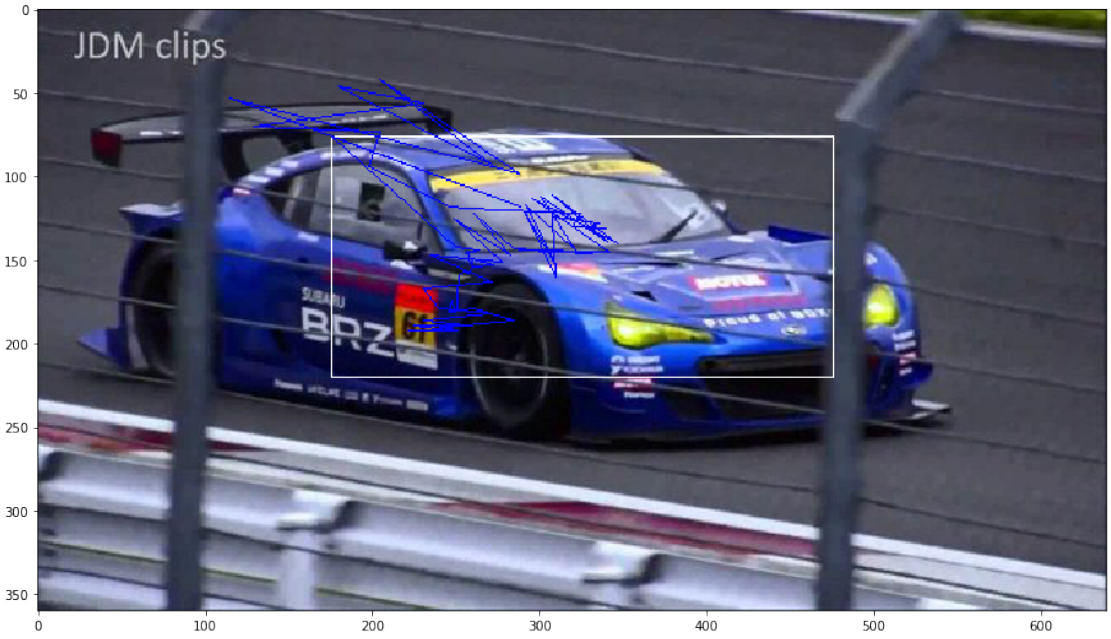

JDM clips
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

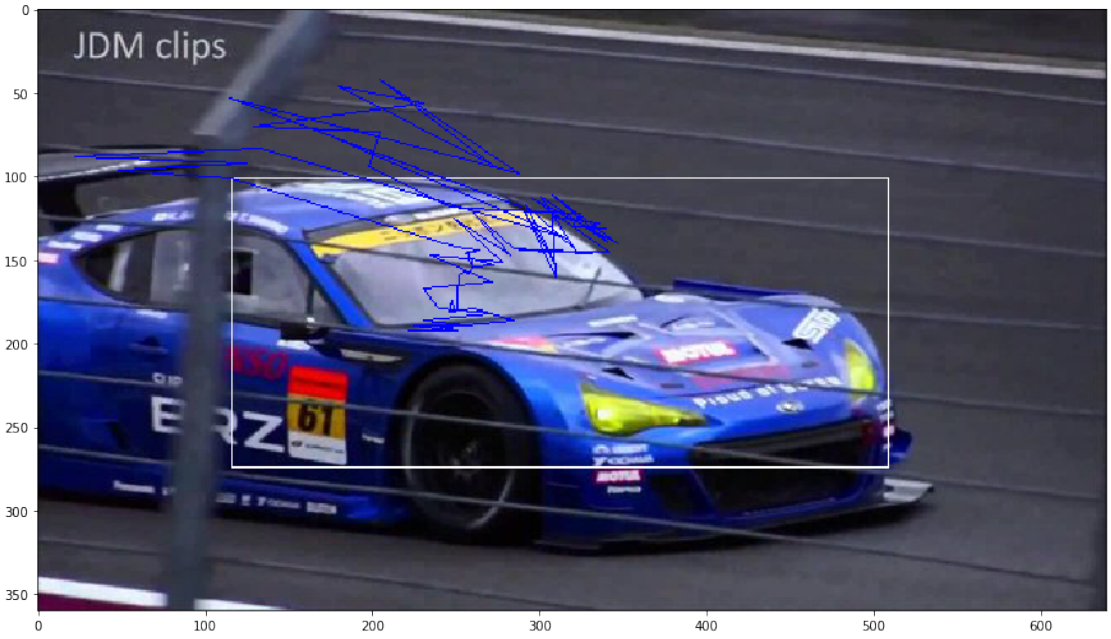

JDM clips
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

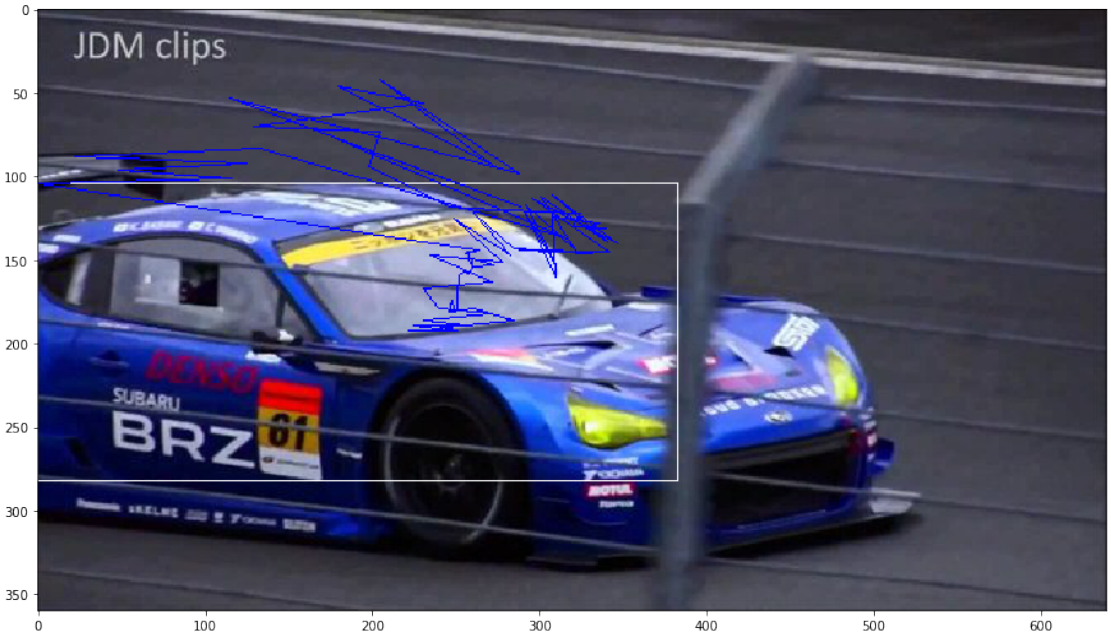

JDM clips
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

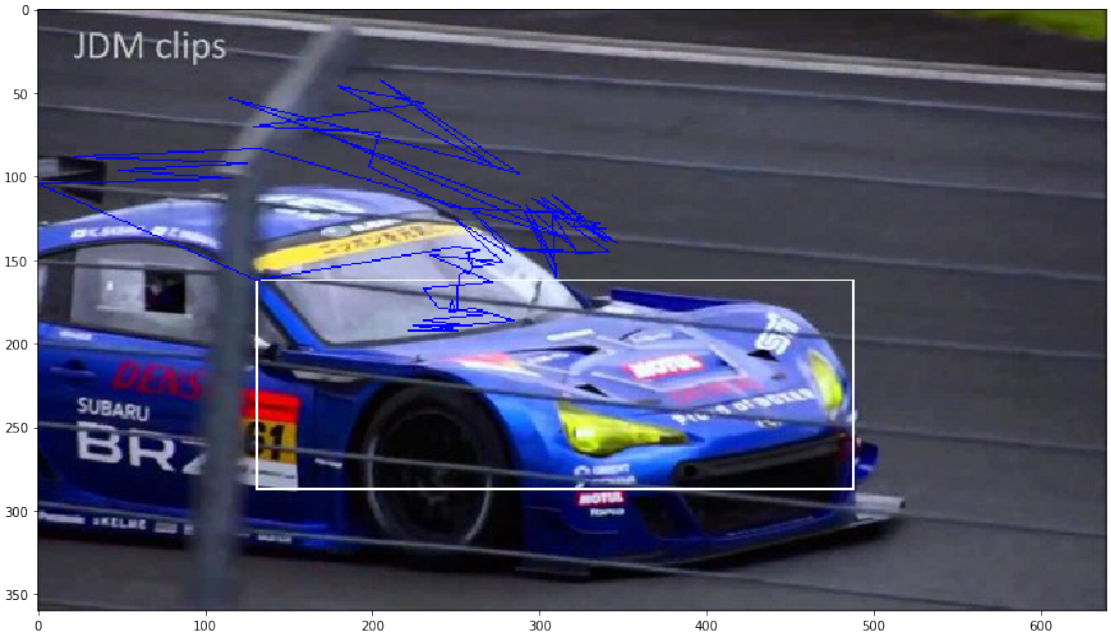

JDM clips
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

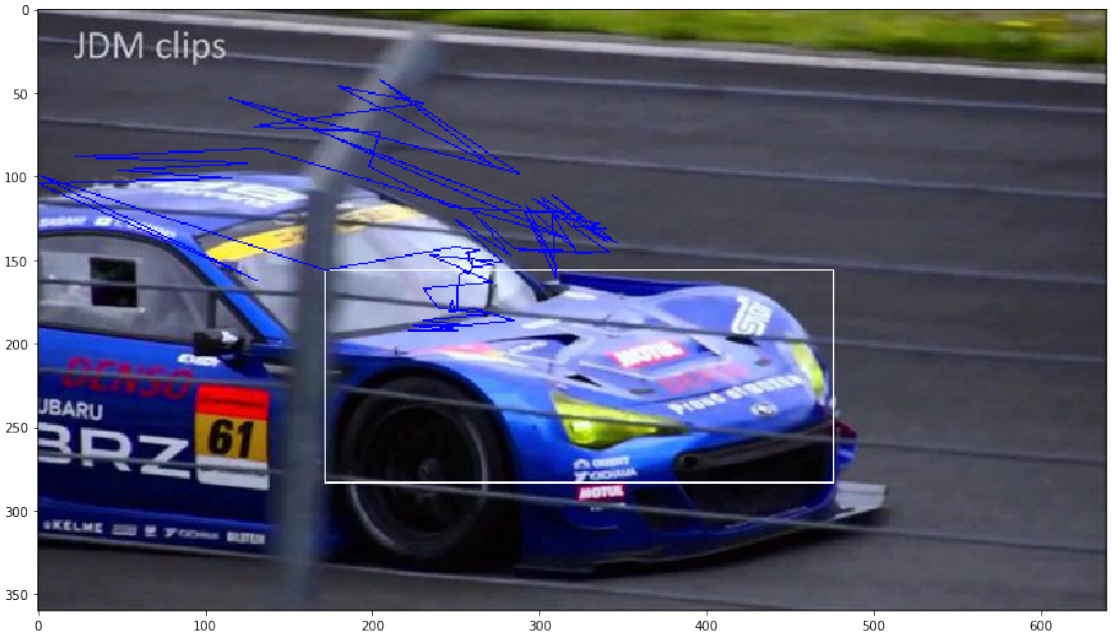

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

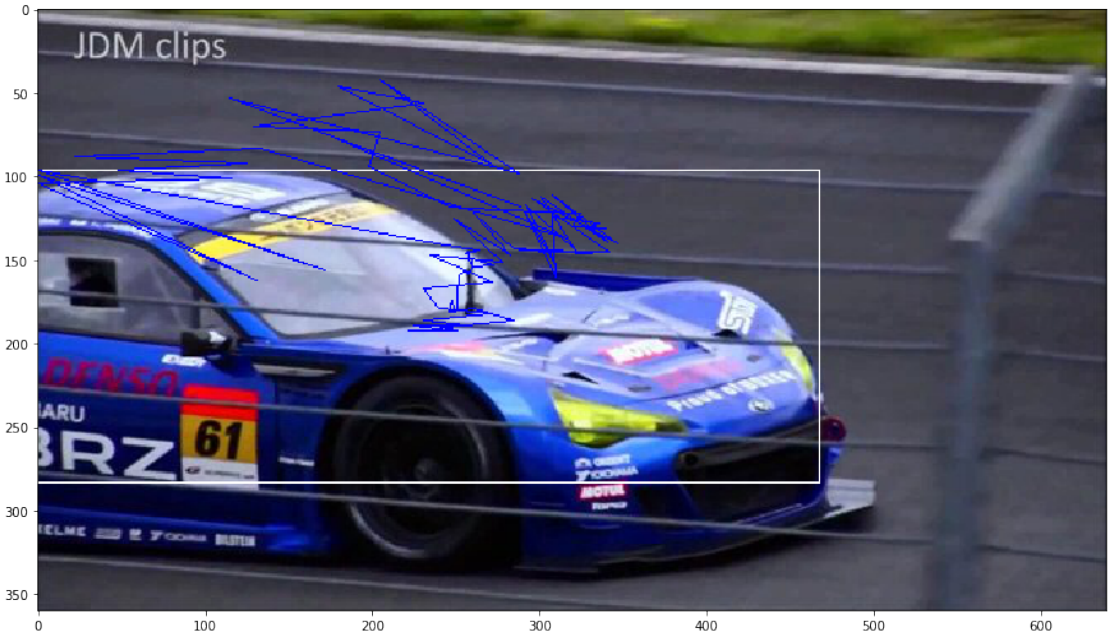

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

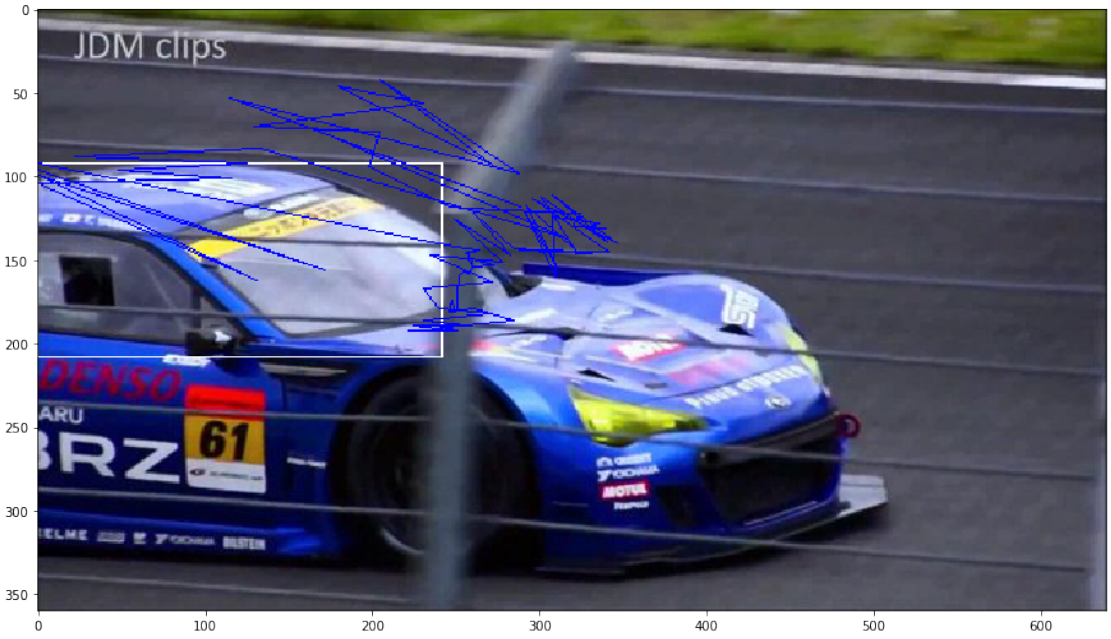

JDM clips
DENSO
SUBARU
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

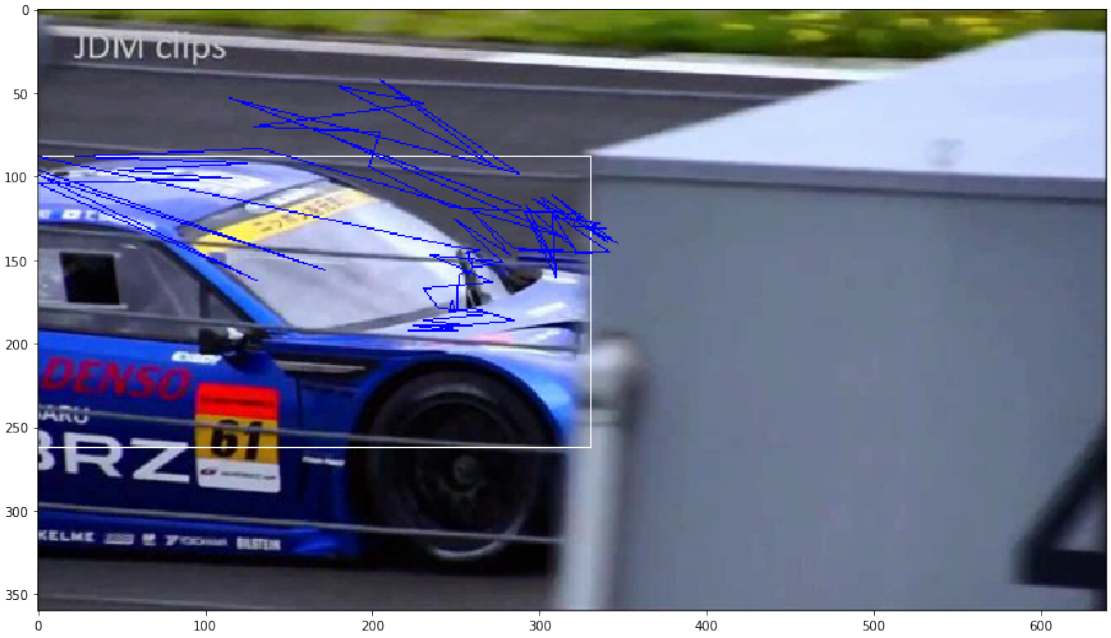

JDM clips
SUBARU
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

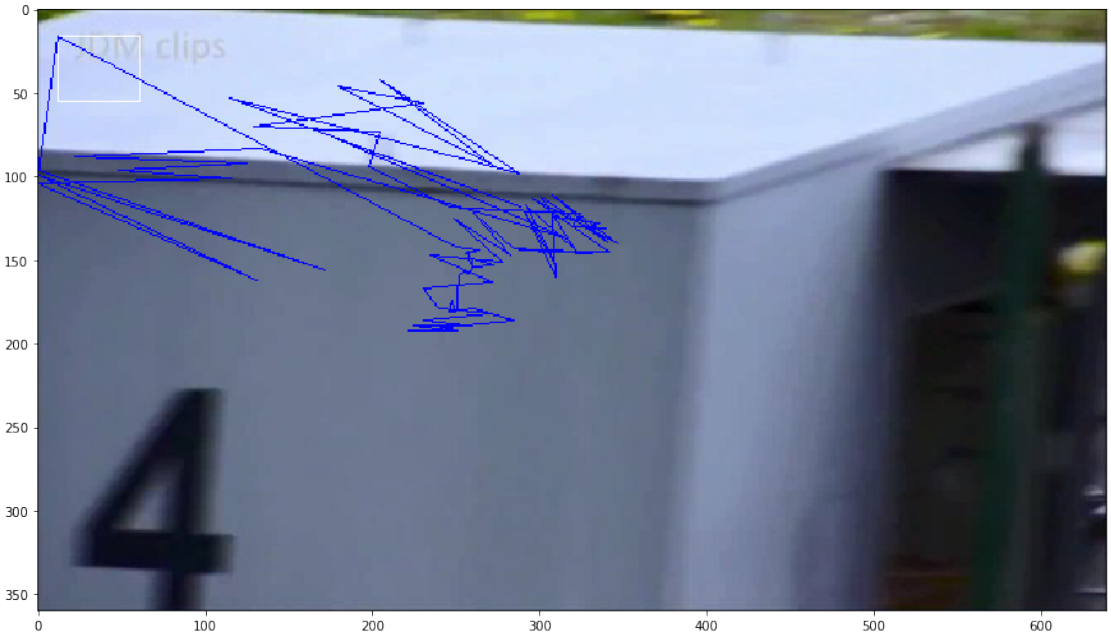

JDM clips
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

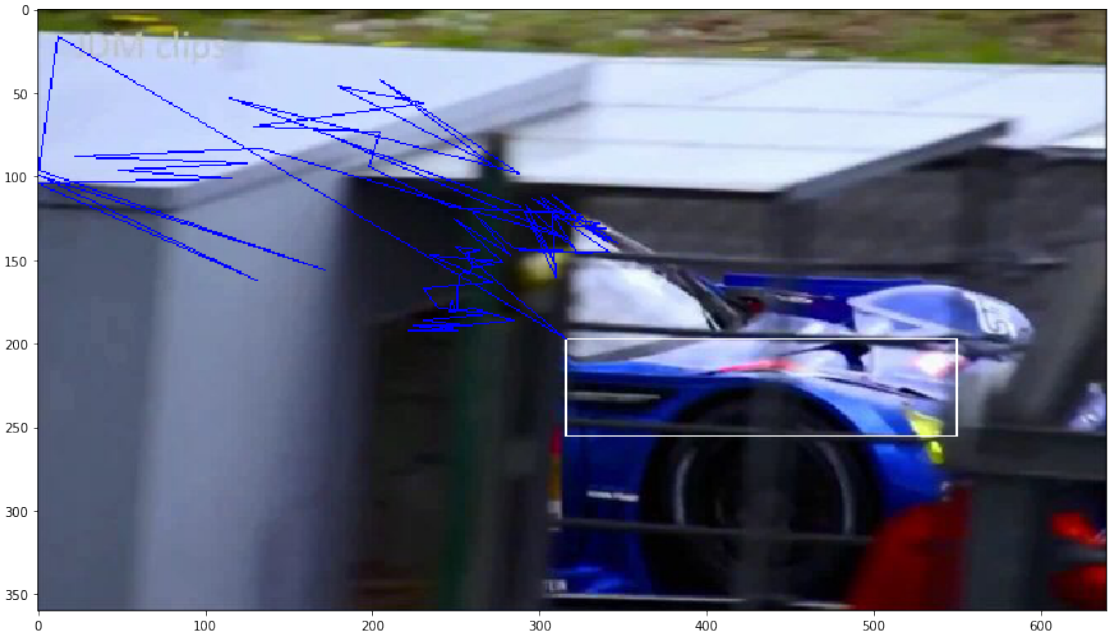

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

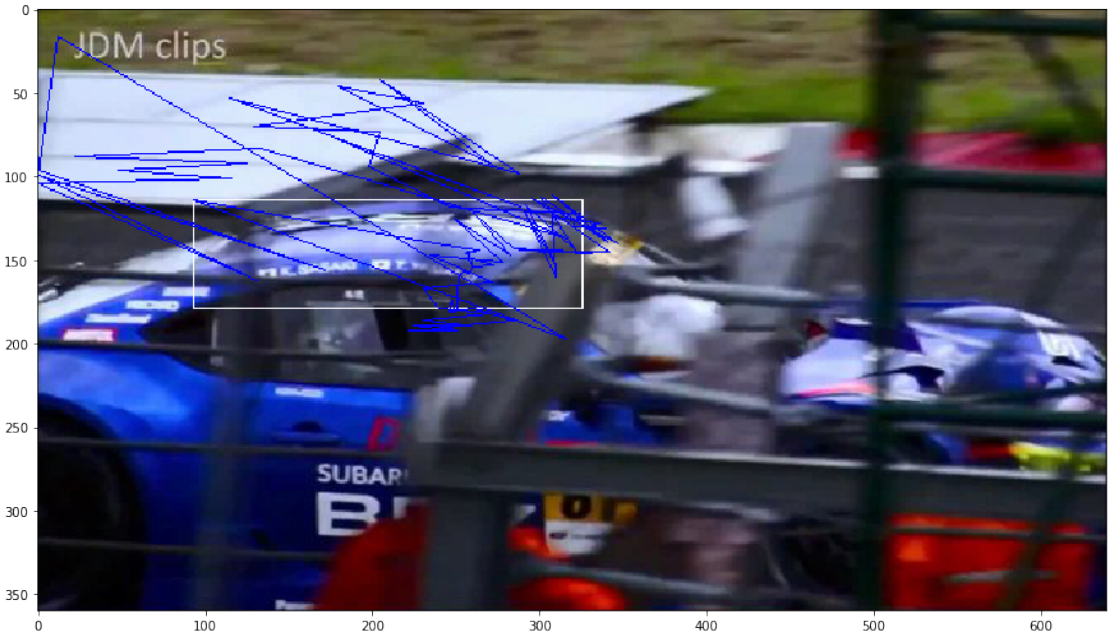

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
DENSO
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

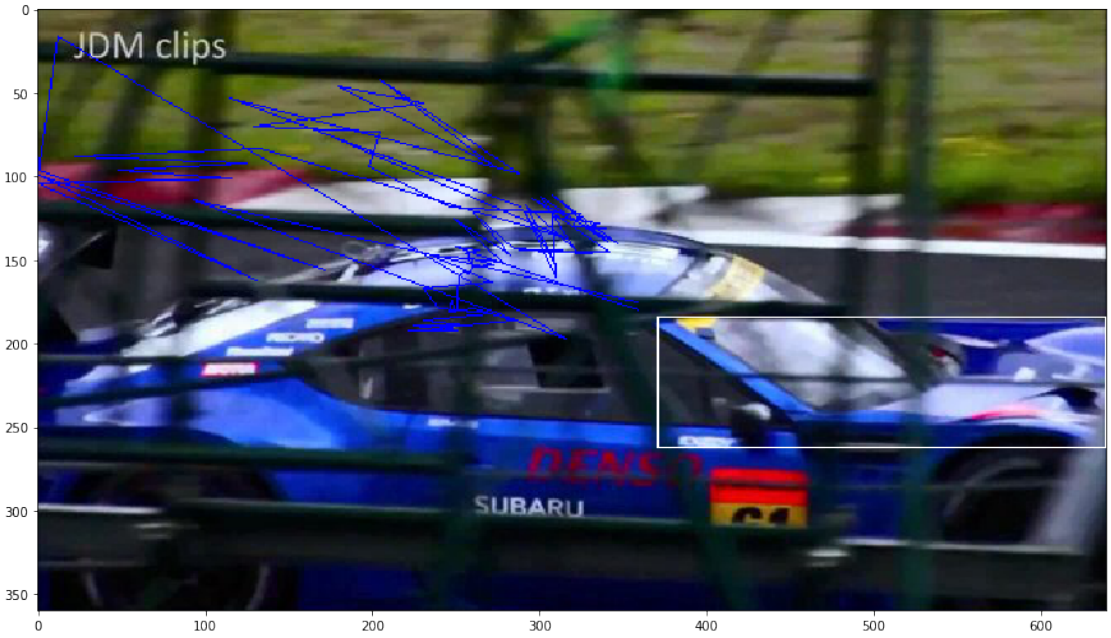

JDM clips
4
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

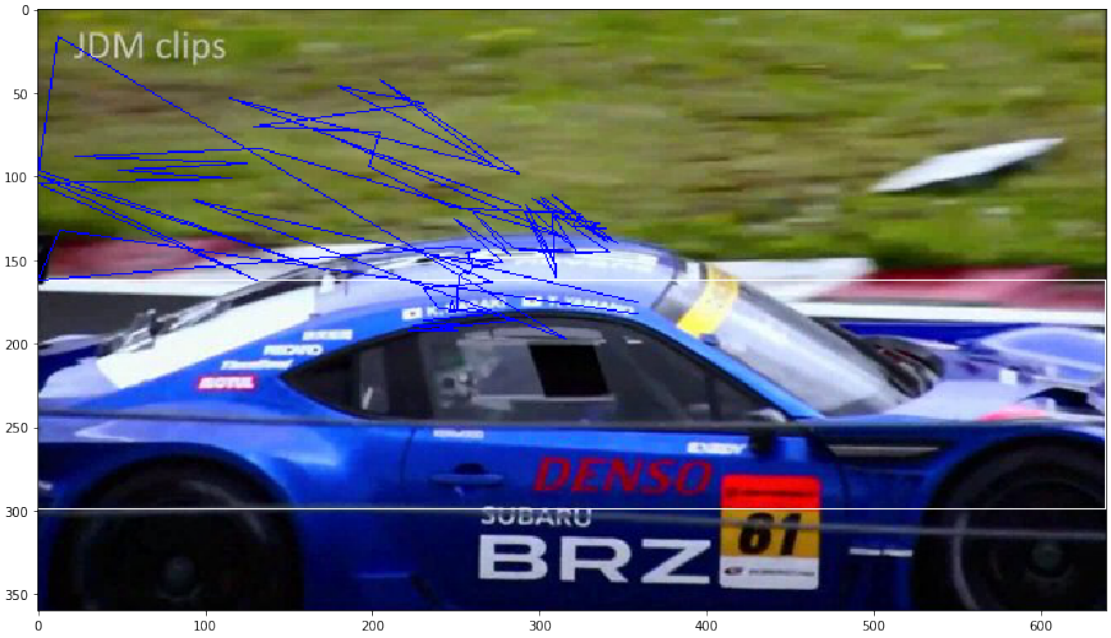

0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

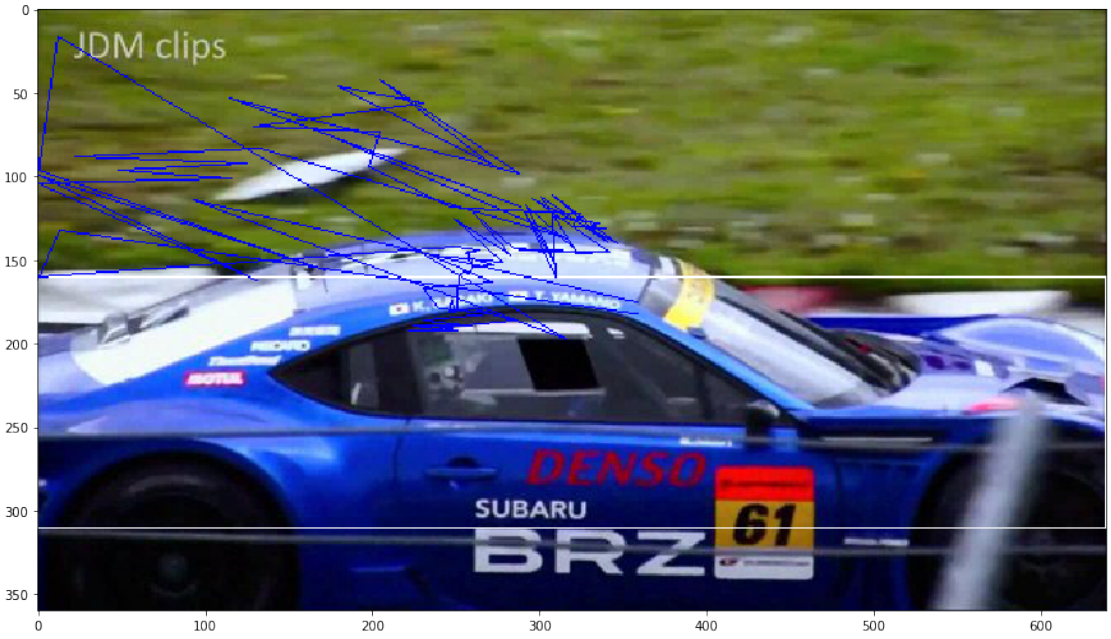

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

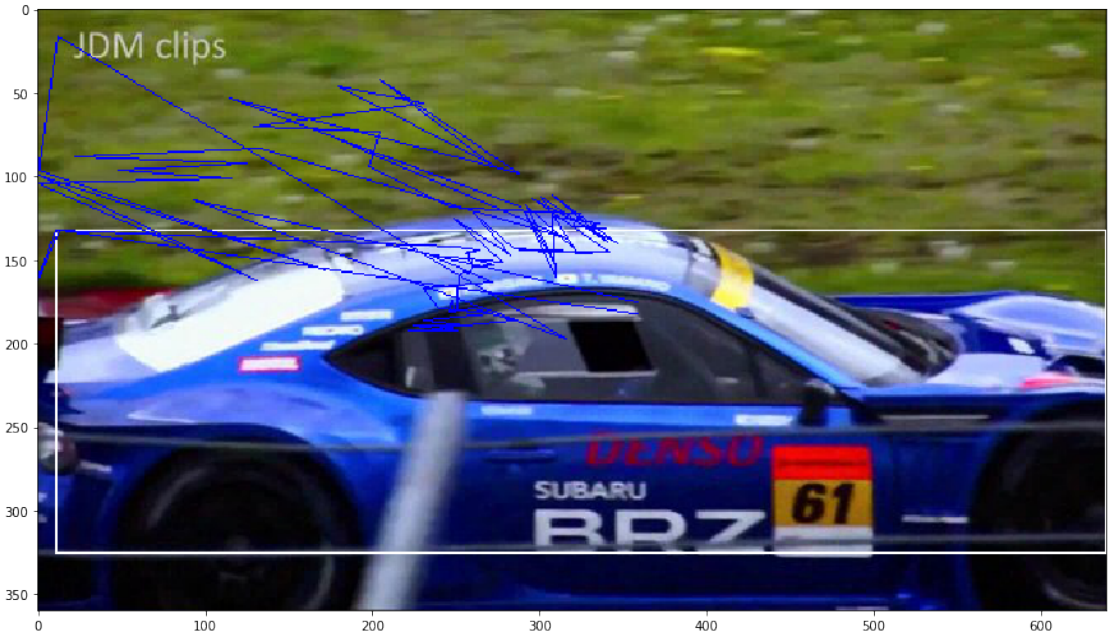

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
SUBARU
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

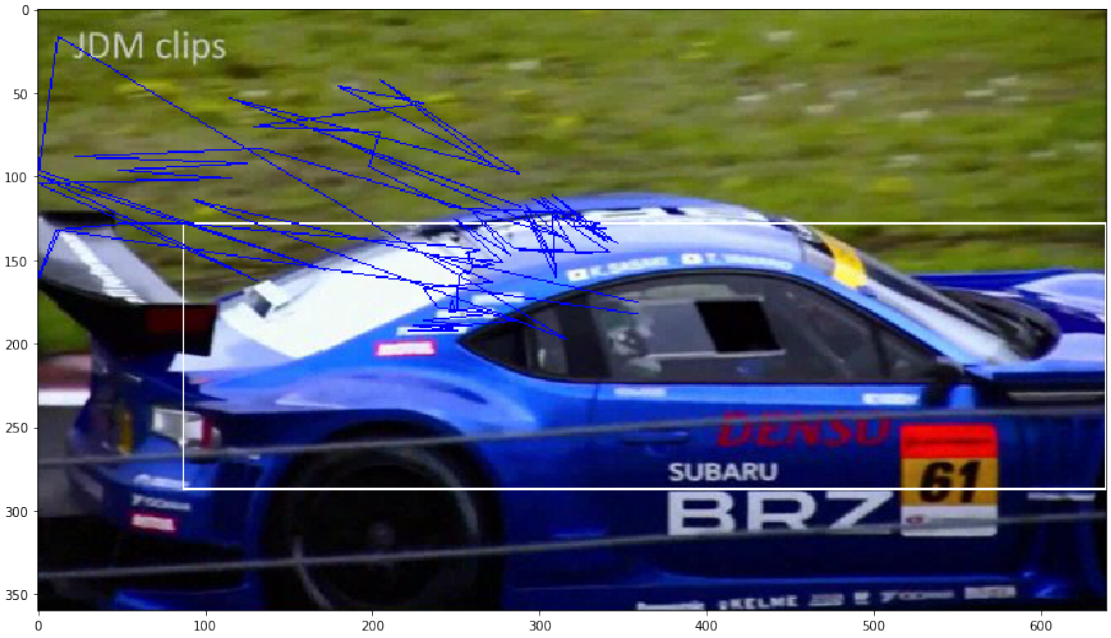

JDM clips
DENSO
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

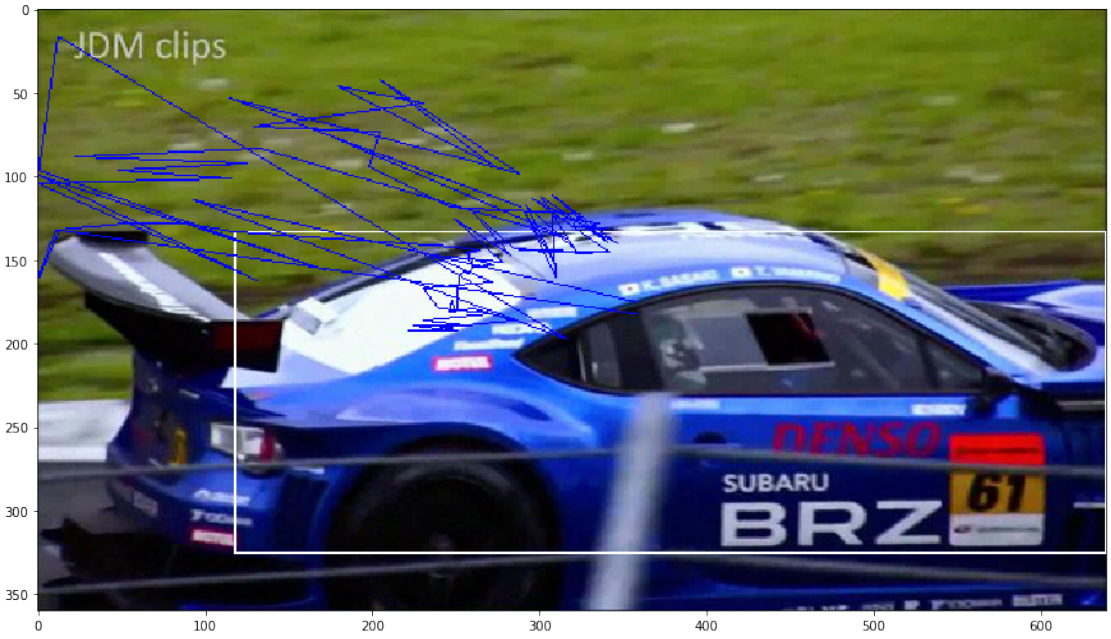

JDM clips
DENSO
SUBARU
BRZ
61

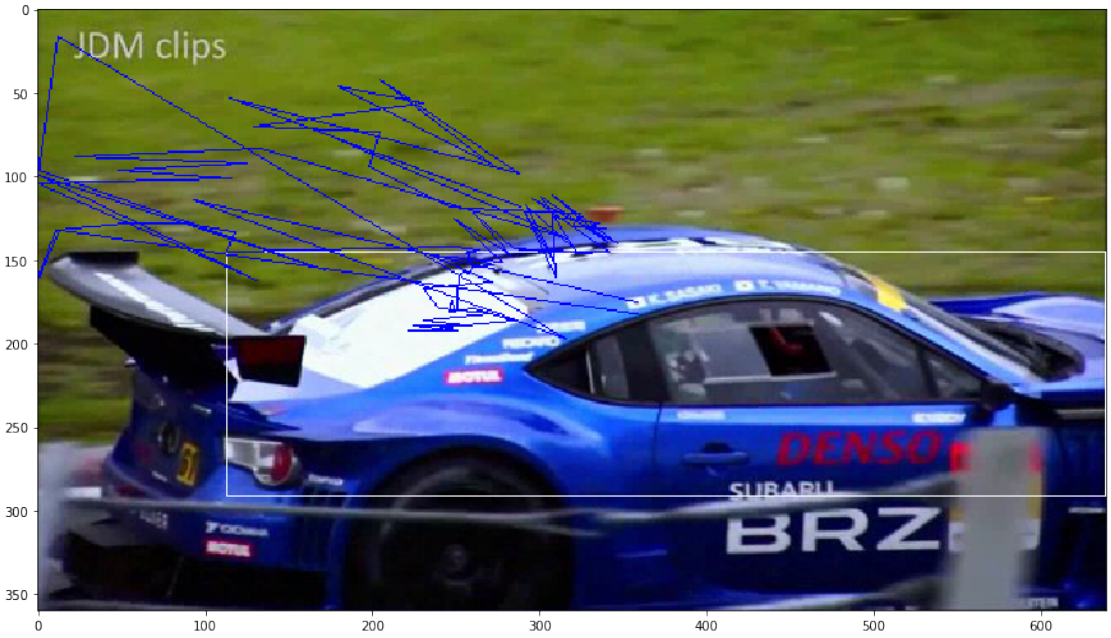

JDM clips
DENSO
SUBARU
BRZ
61

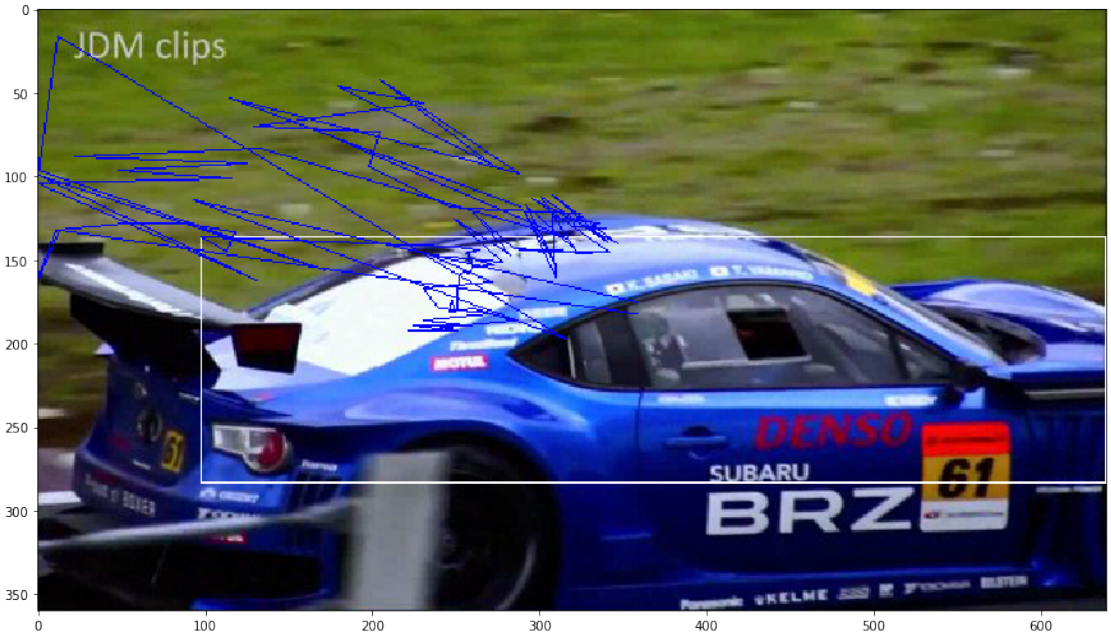

JDM clips
DENSO
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


JDM clips
DENSO
SUBARU
BRZ
61
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

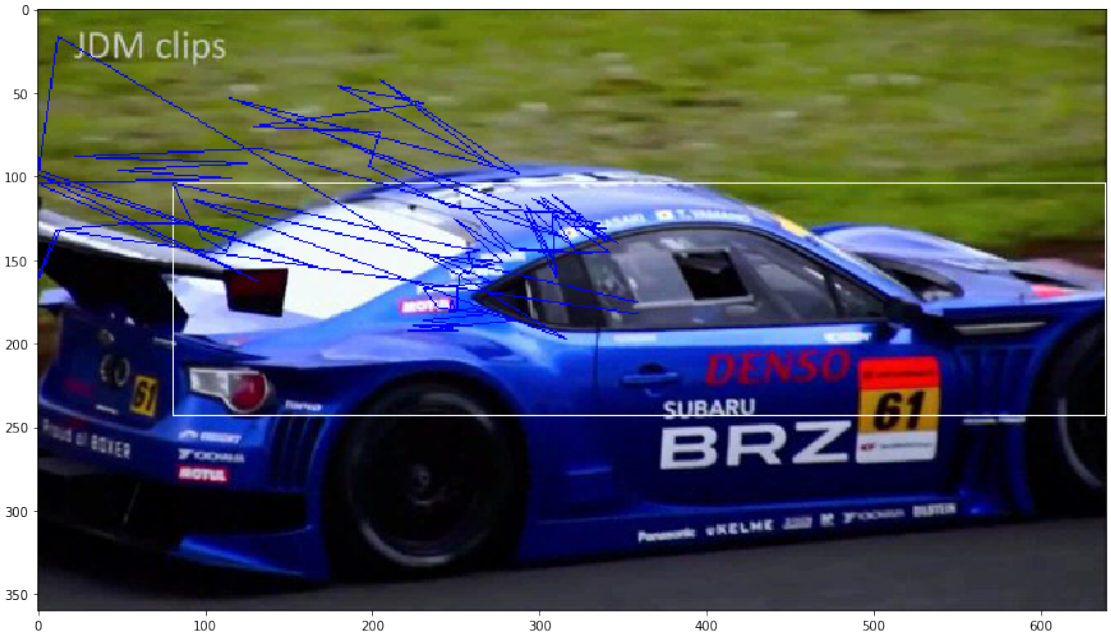

JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ


JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ
61

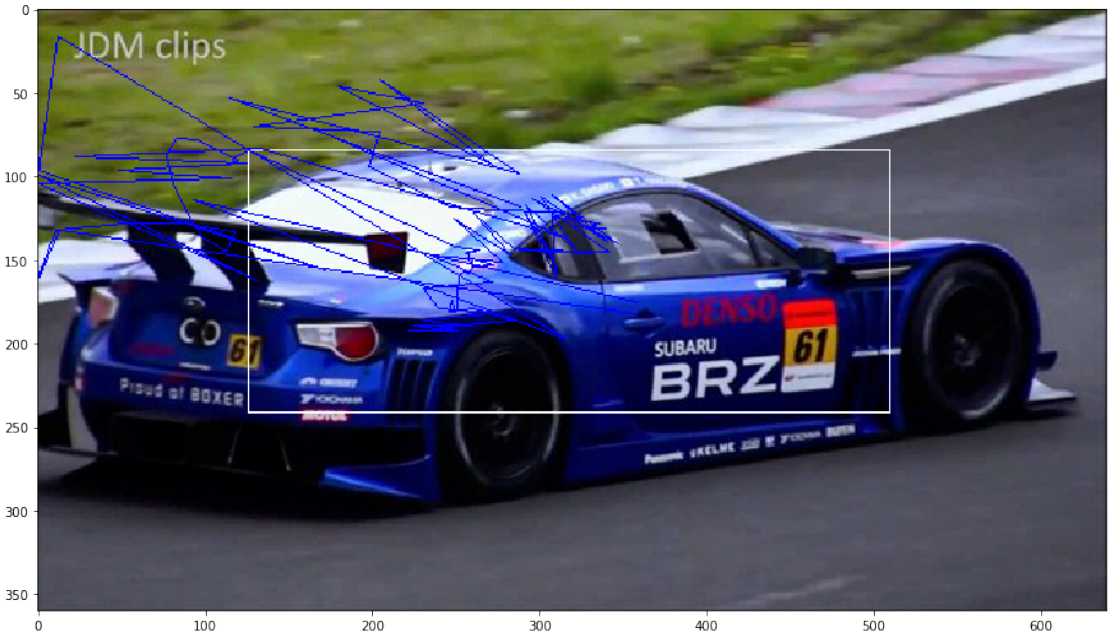

JDM clips
DENSO
SUBARU
BRZ
61


JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER
MOTUL
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

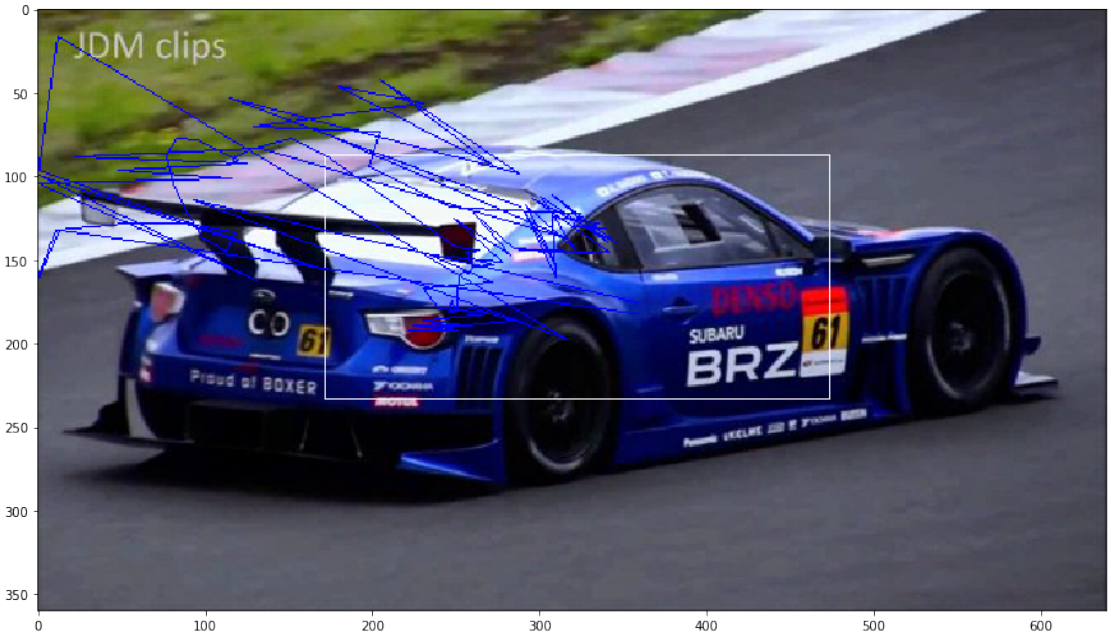

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER
MOTUL
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

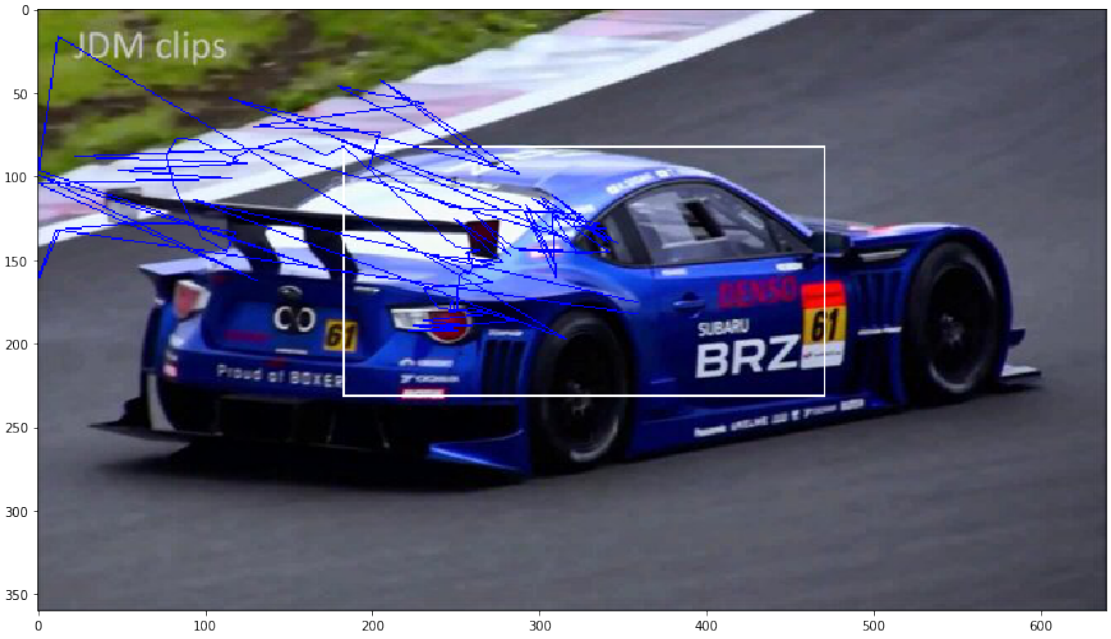

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

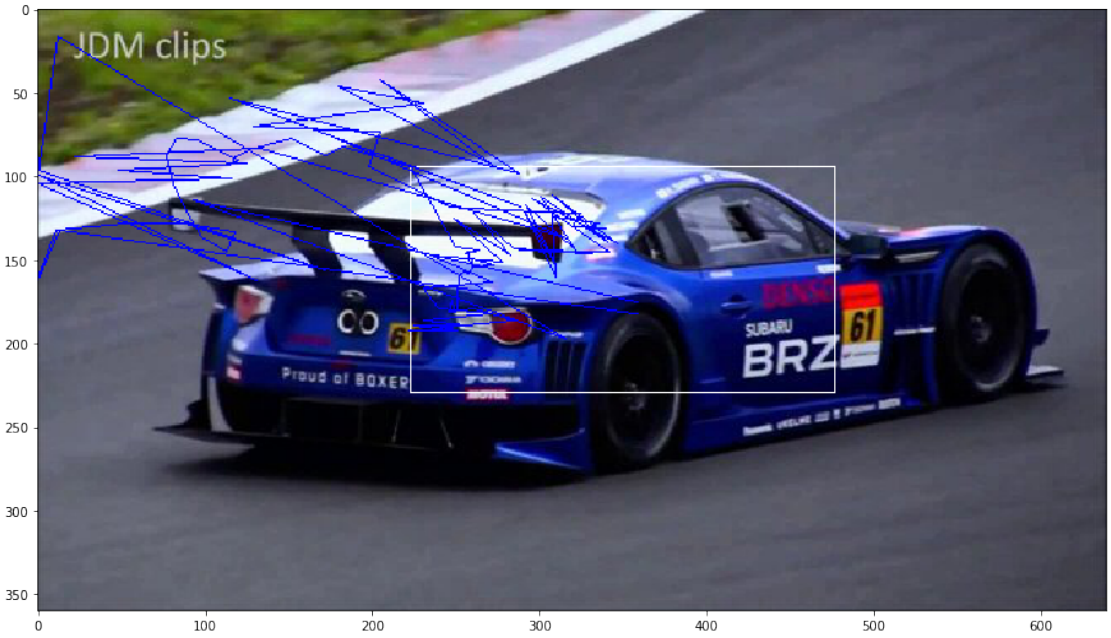

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

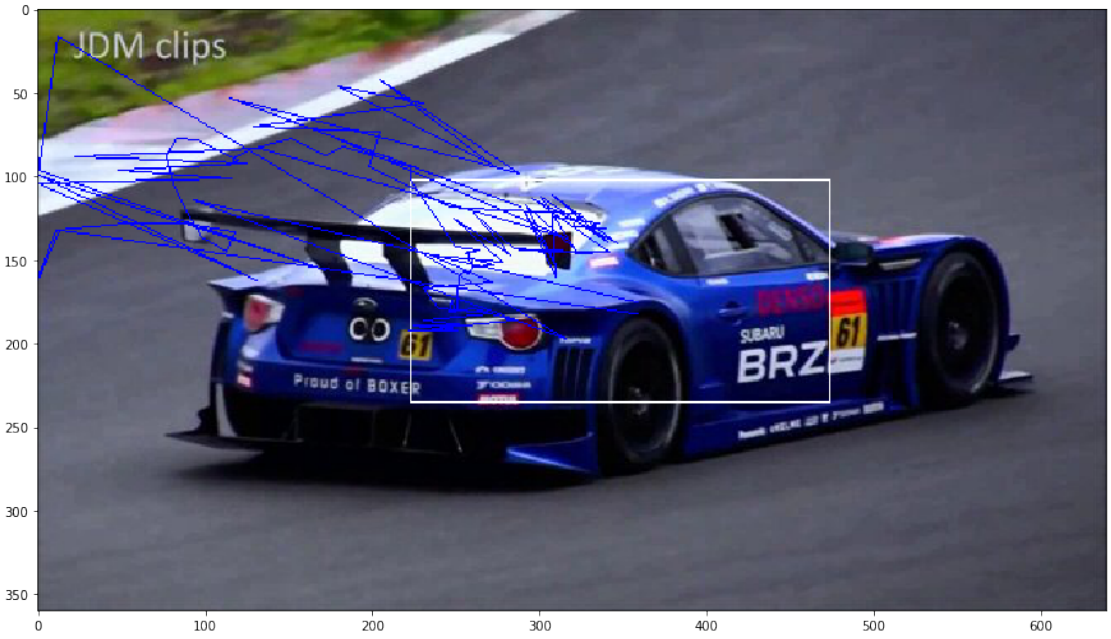

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER

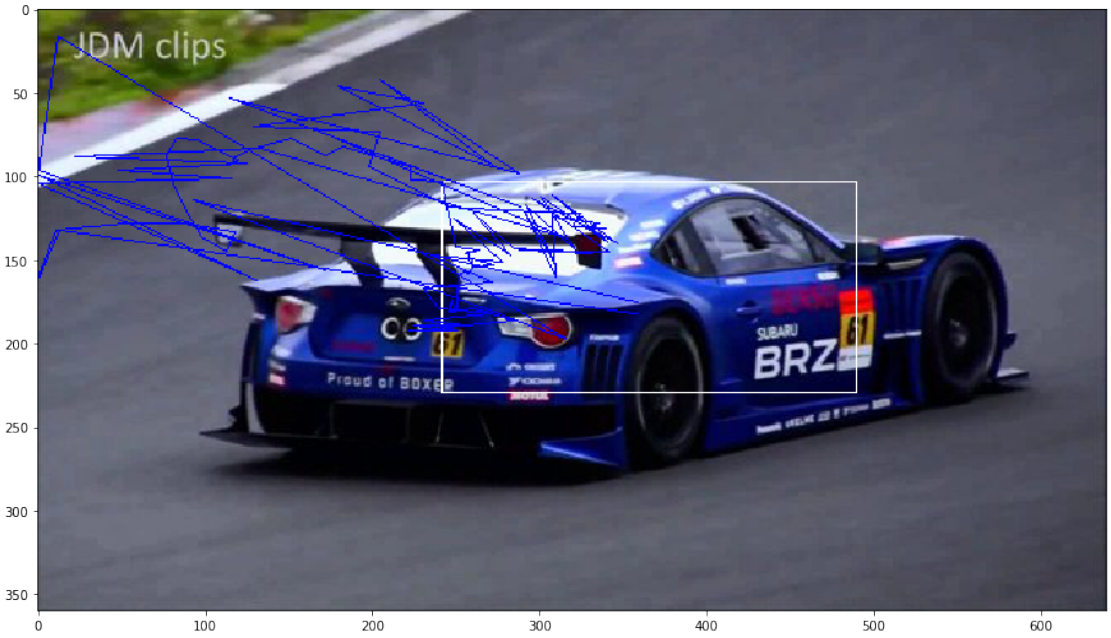

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER

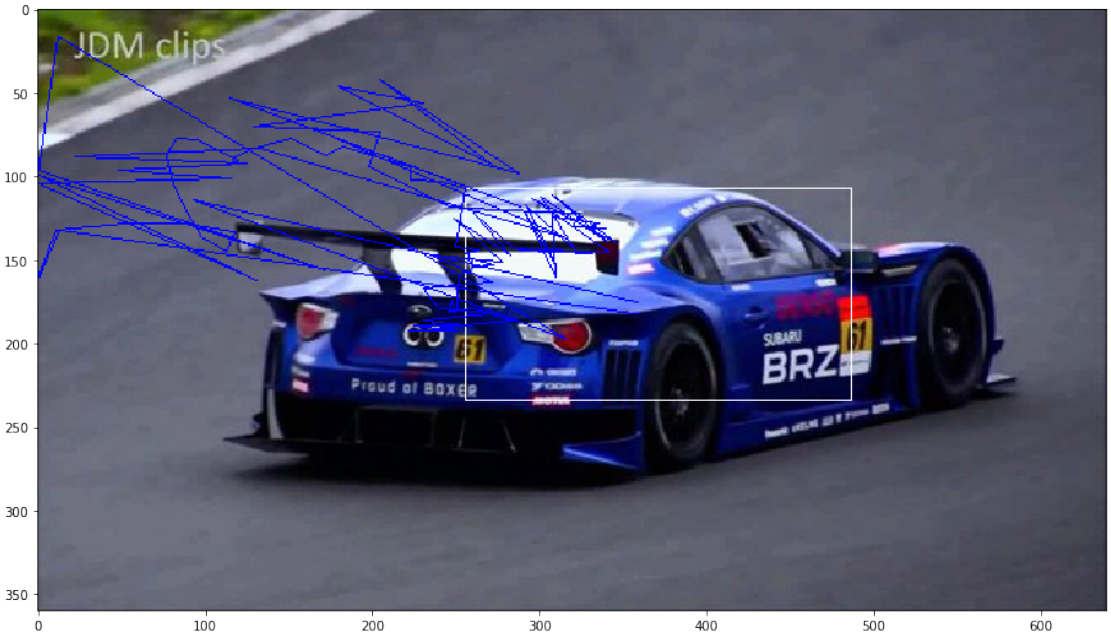

JDM clips
DENSO
SUBARU
BRZ
61
Proud of BOXER

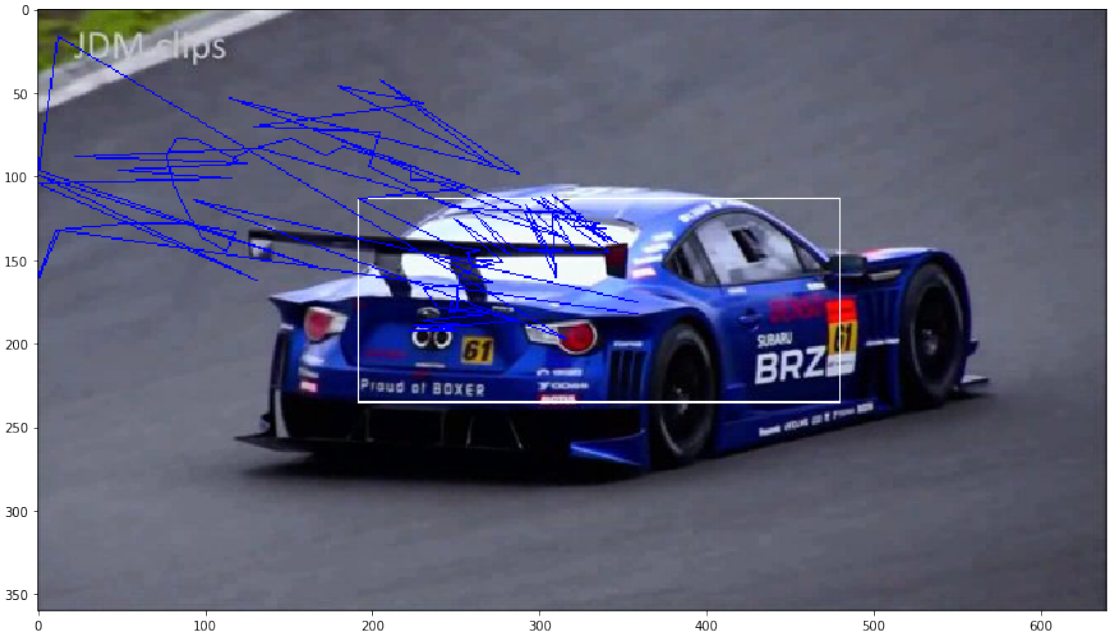

JDM clips
DENSO
SUBARU
BRZ
61

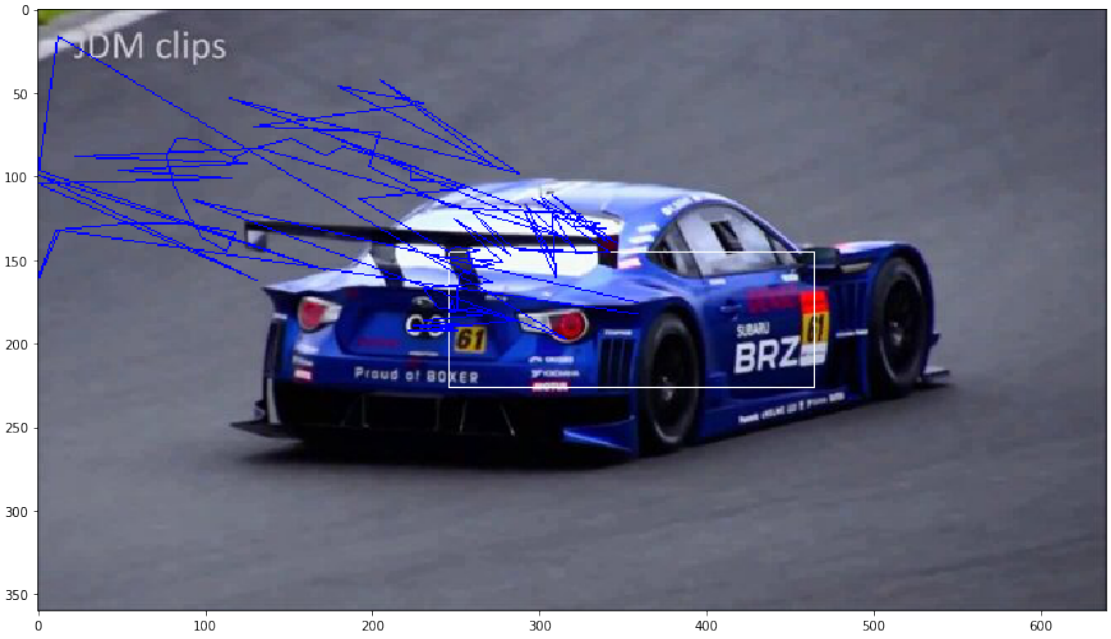

JDM clips
SUBARU
BRZ
61
Proud of BOXER

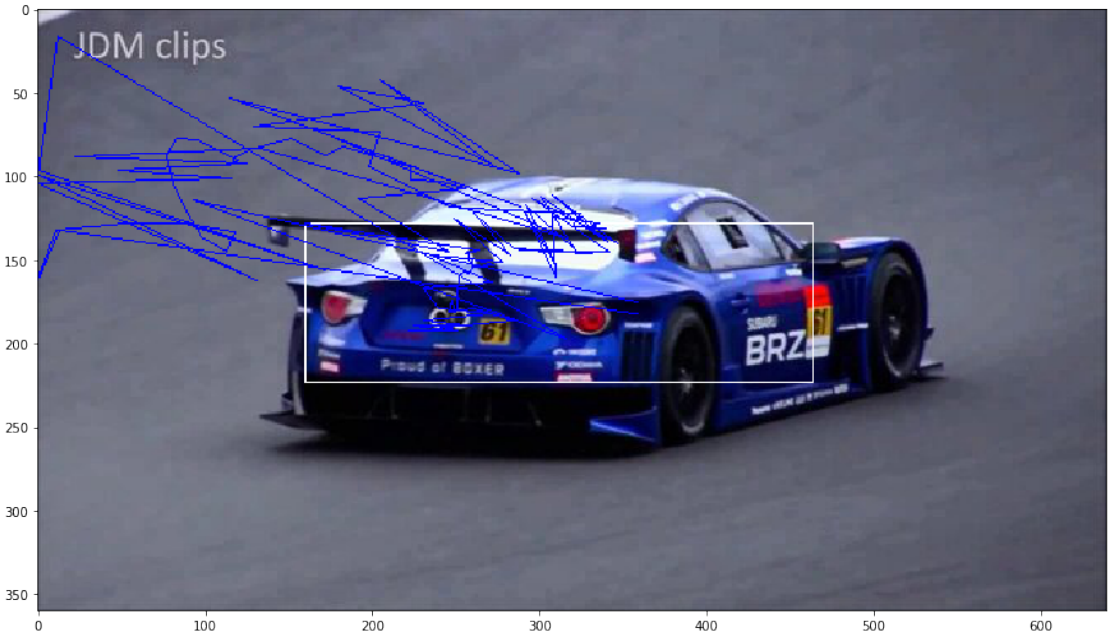

JDM clips
SUBARU
BRZ
61
Proud of BOXER

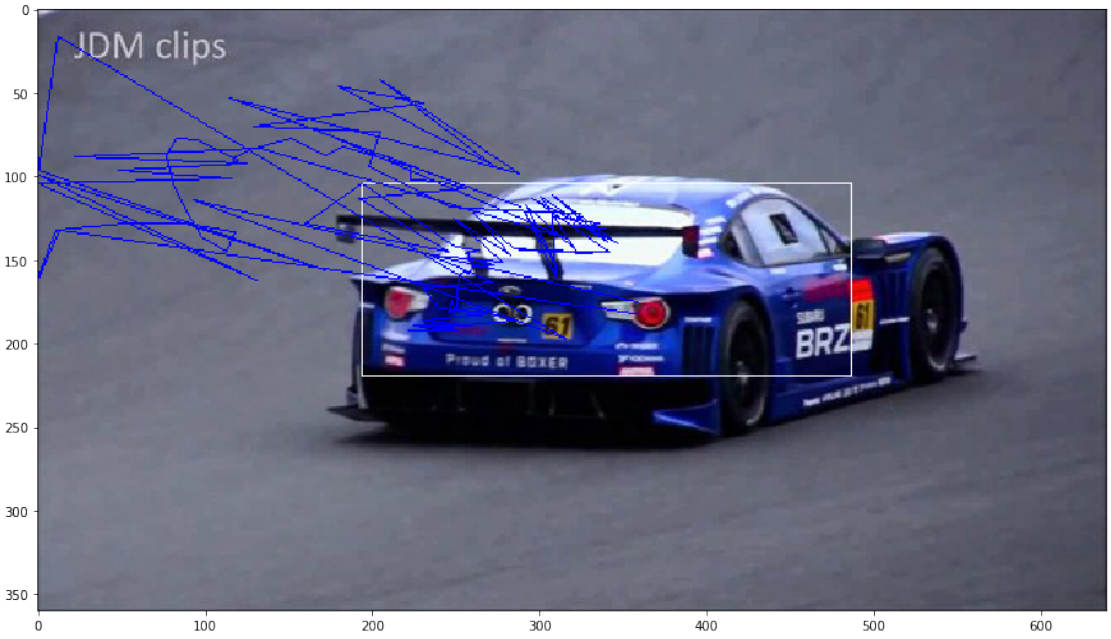

JDM clips
SUBARU
BRZ
Proud of Boxer
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

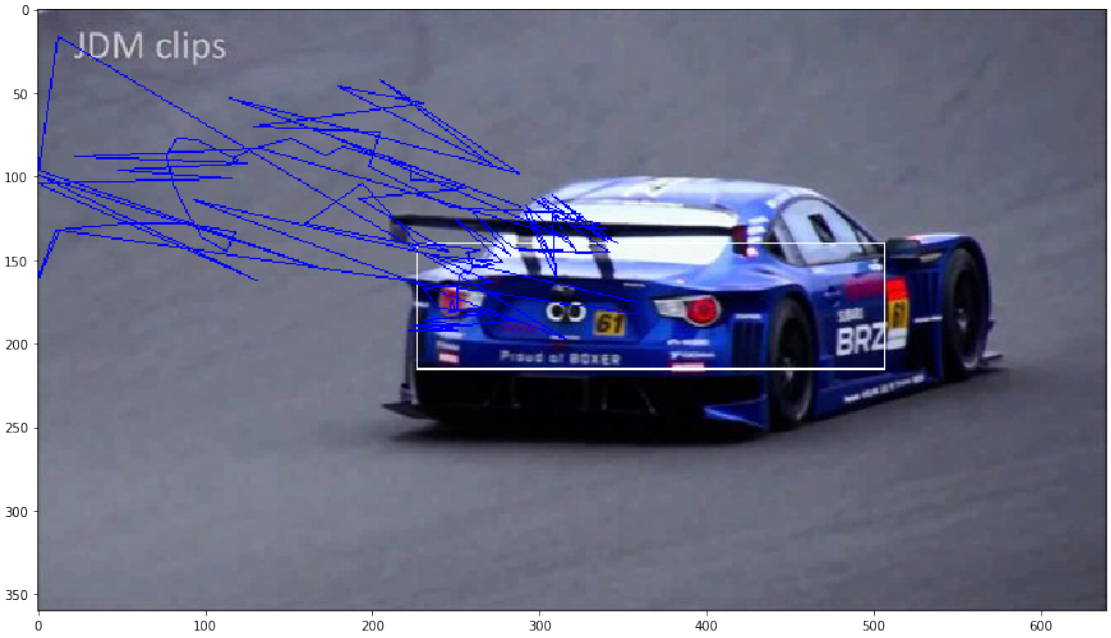

JDM clips
SUBARU
BRZ
Proud of Boxer
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

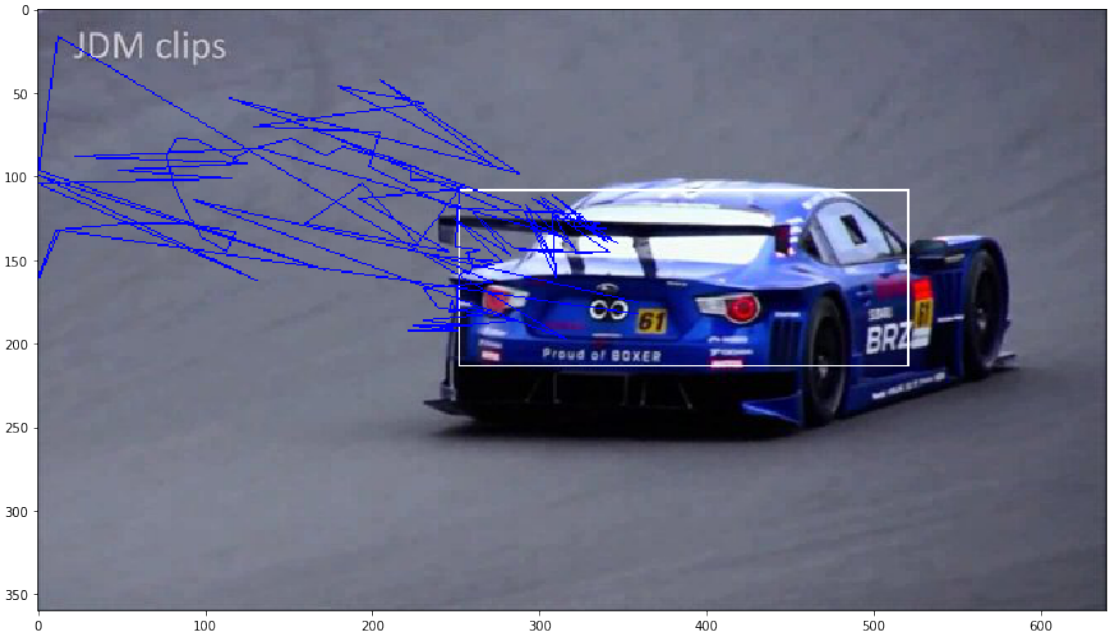

JDM clips
SUBARU
BRZ
61
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

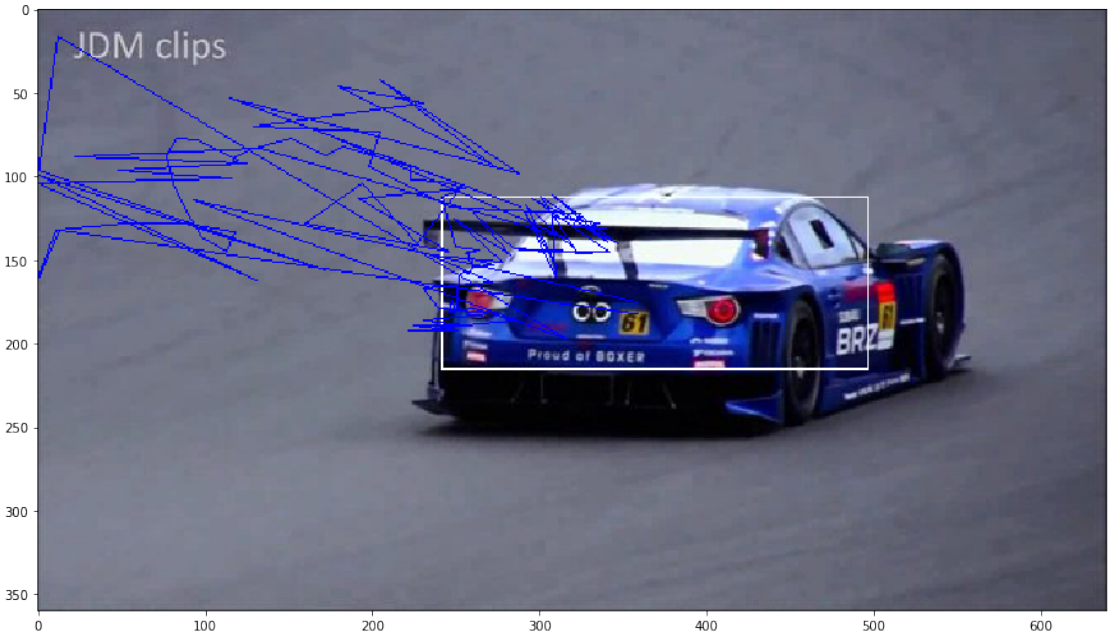

JDM clips
SUBARU
BRZ
61
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

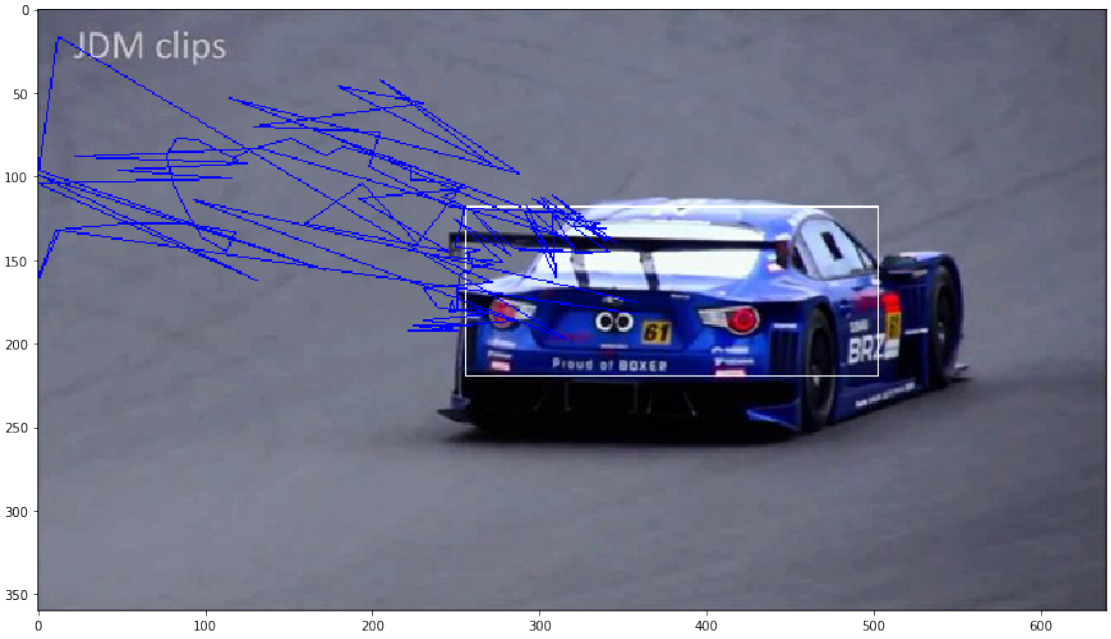

JDM clips
BRZ
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

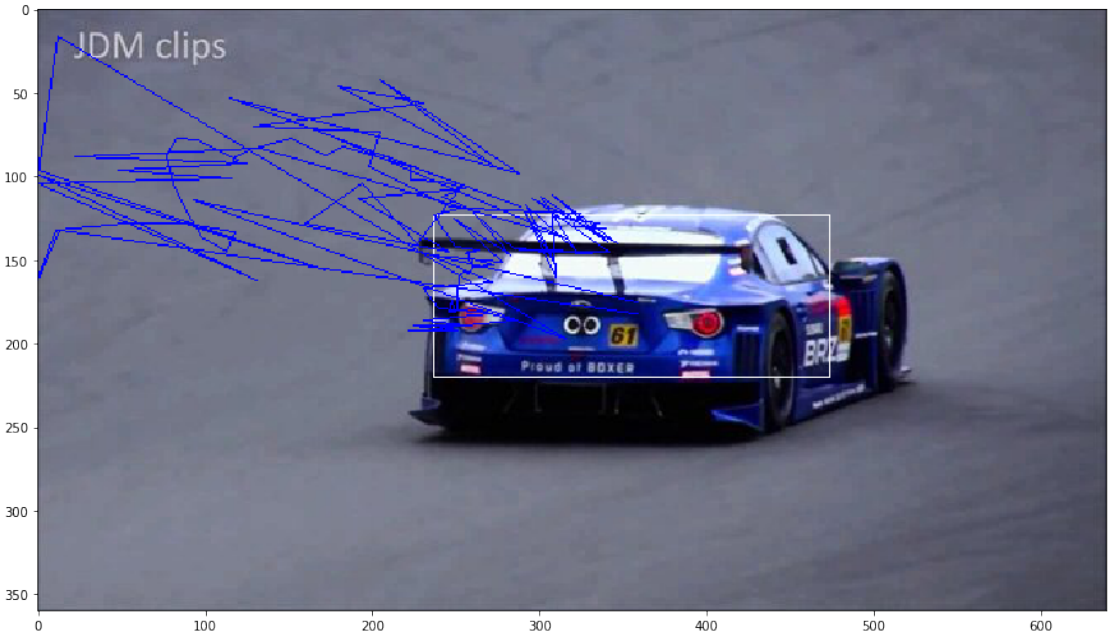

JDM clips
BRZ
Proud of BOXER
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

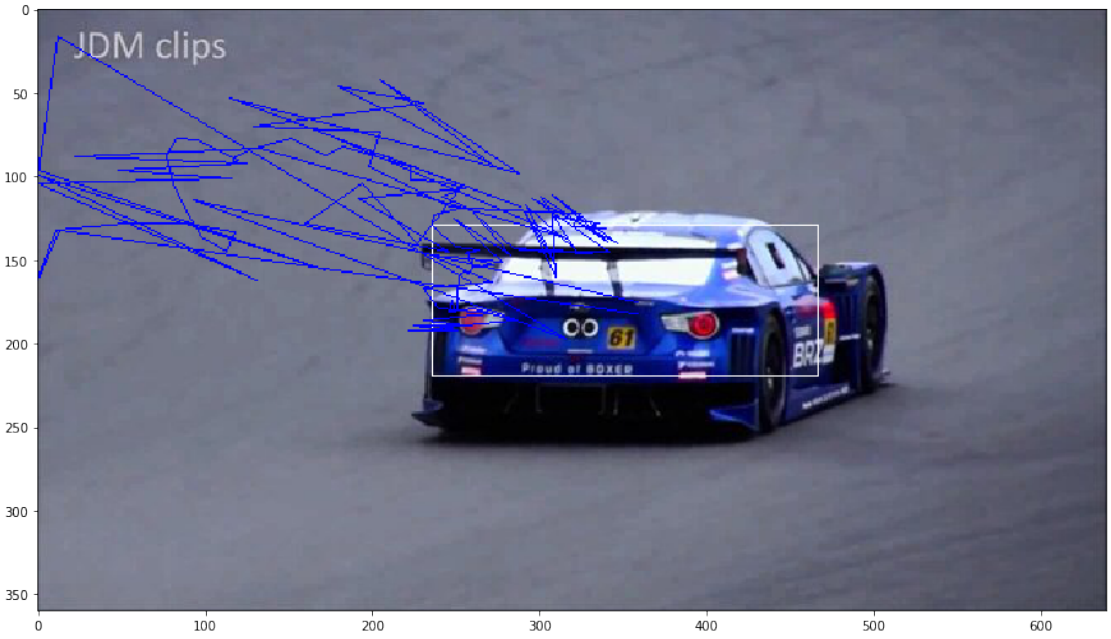

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600
61
BRZ
Proud of BOXER

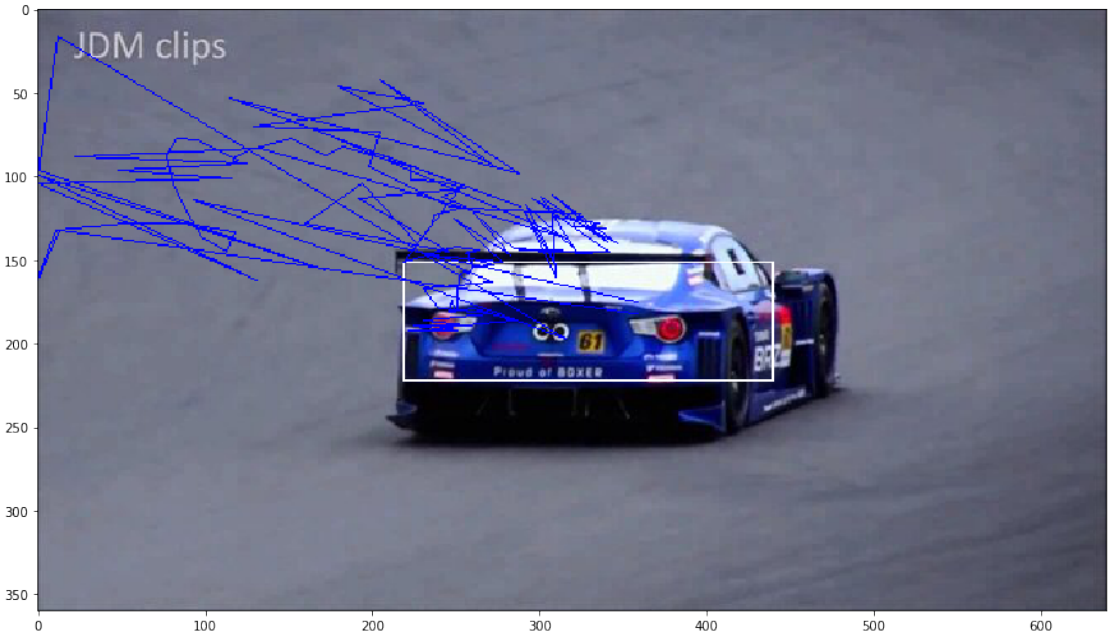

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600
61
BRZ
Proud of BOXER

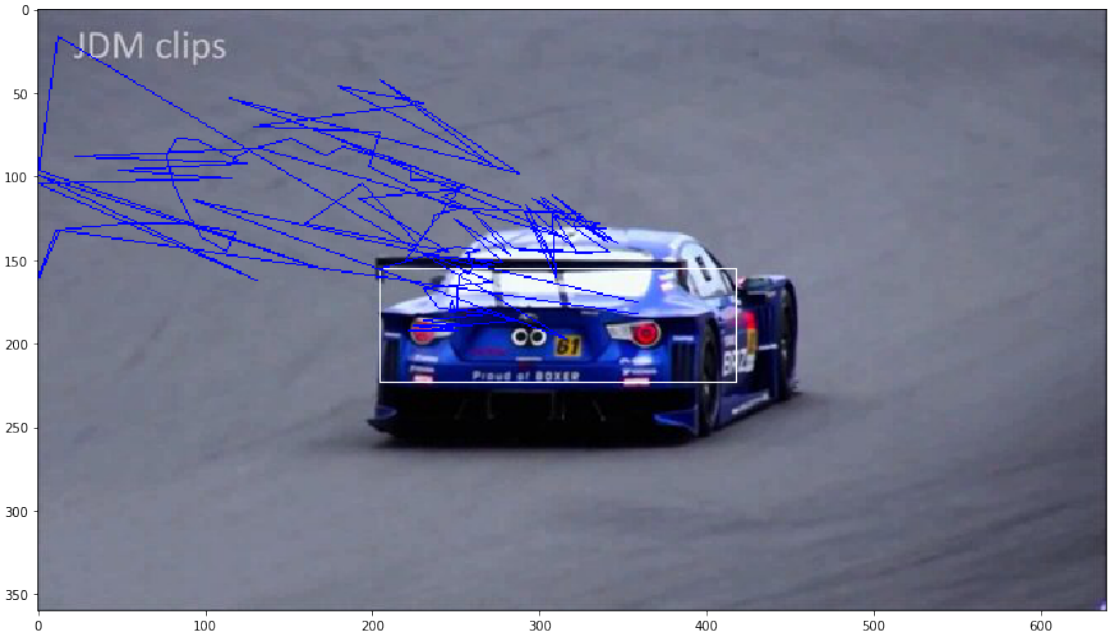

JDM clips
61
Proud of BOXER
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

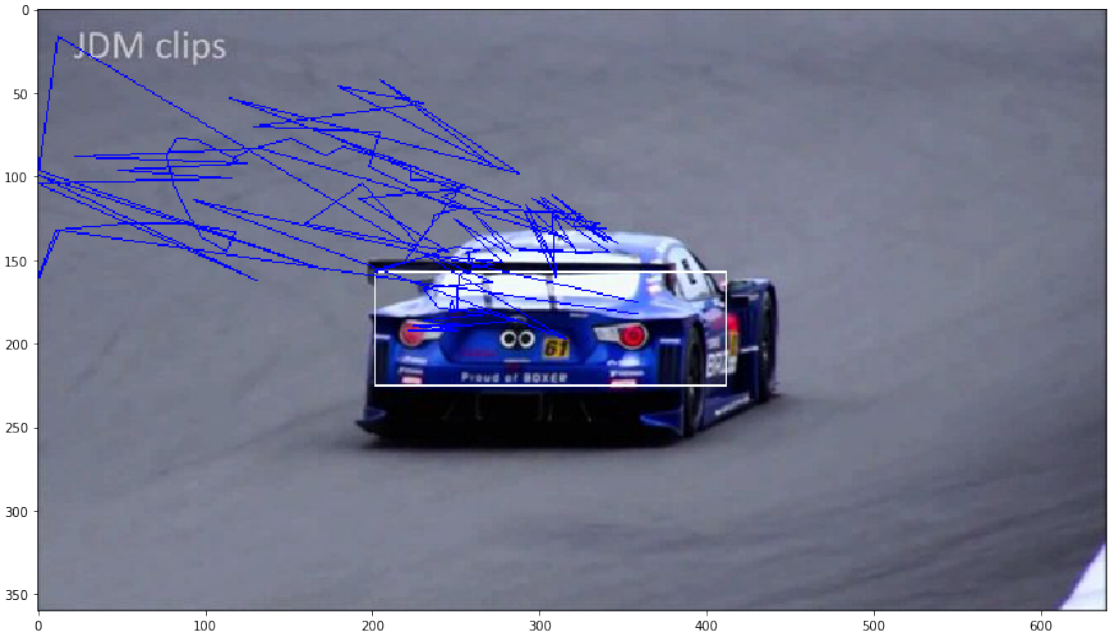

JDM clips
61
Proud of BOXER
BRZ
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

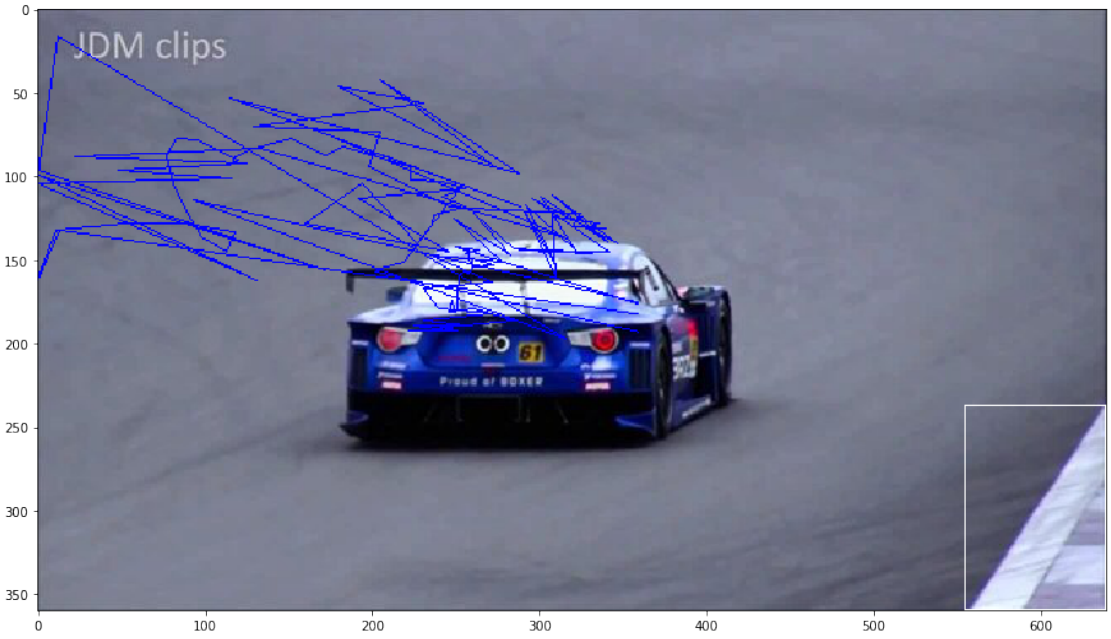

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

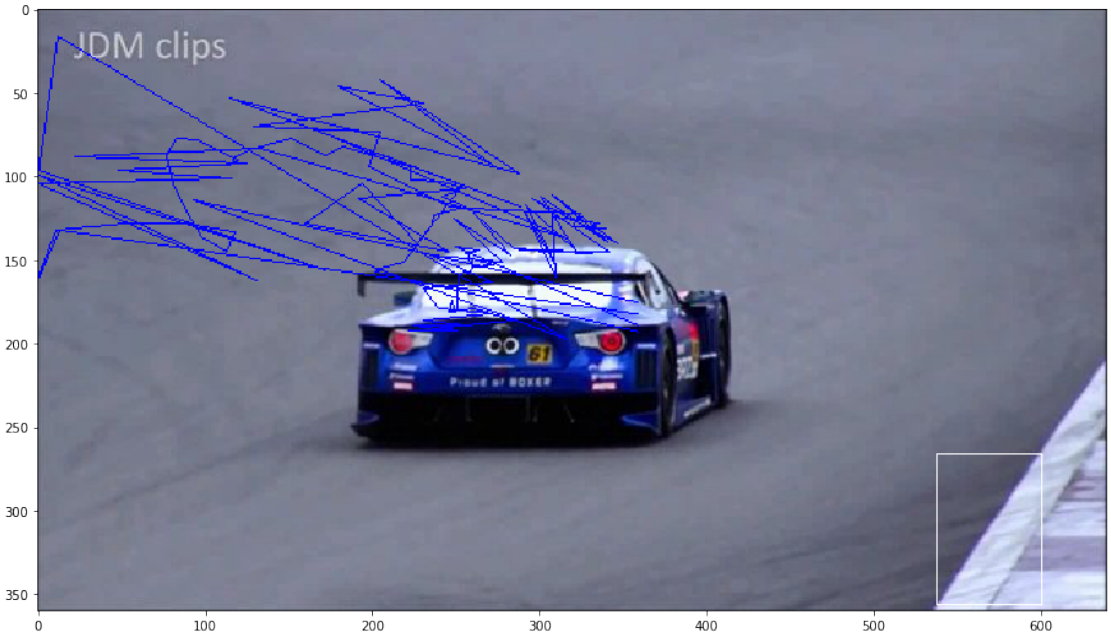

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

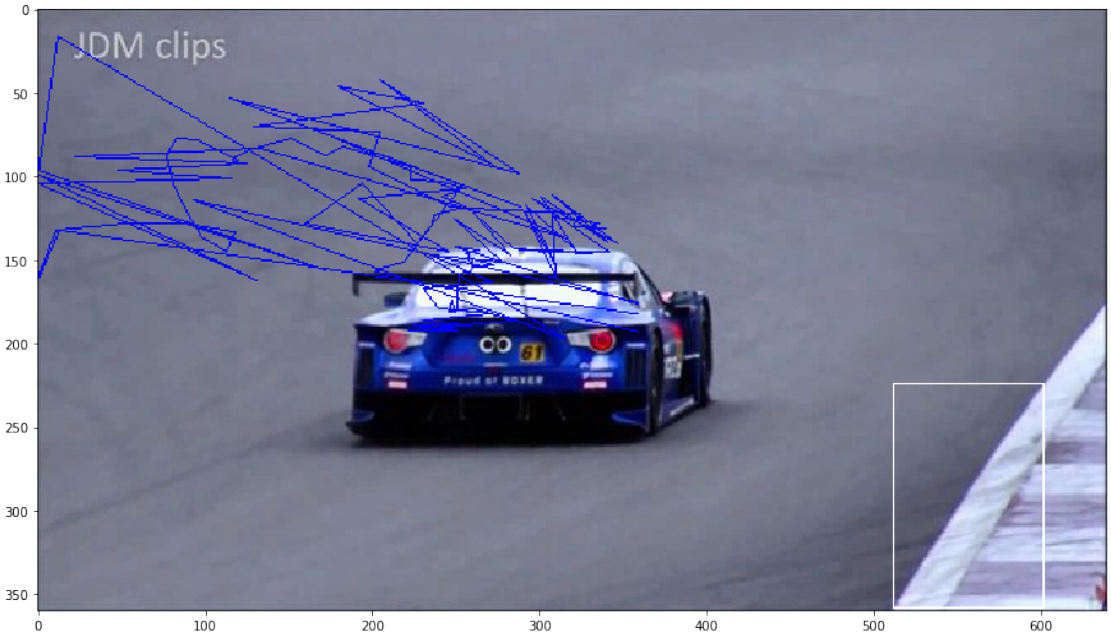

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

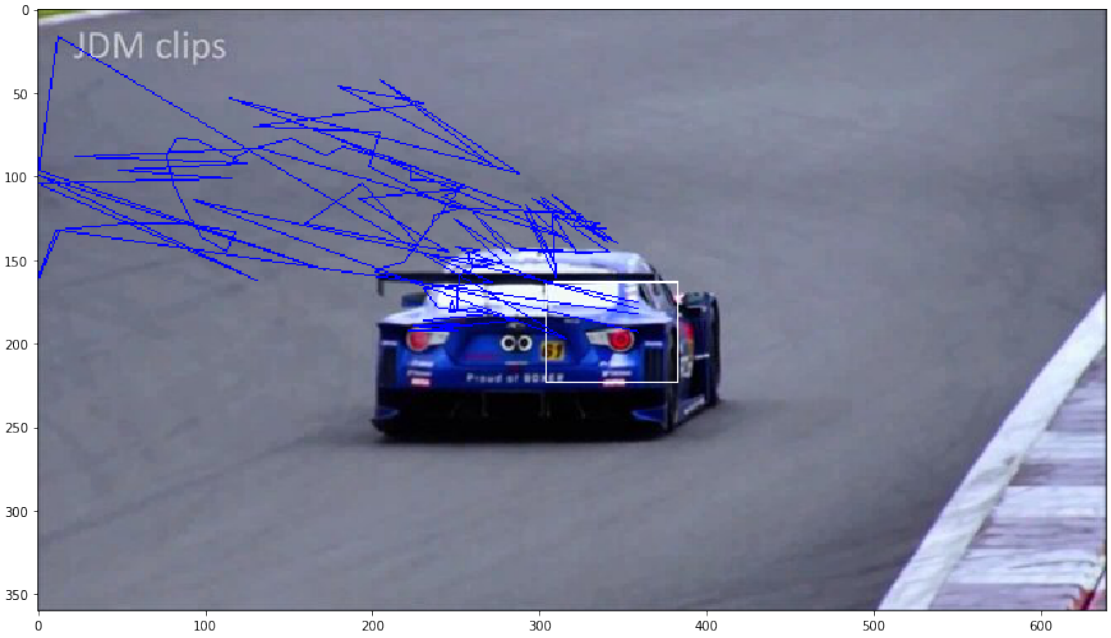

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

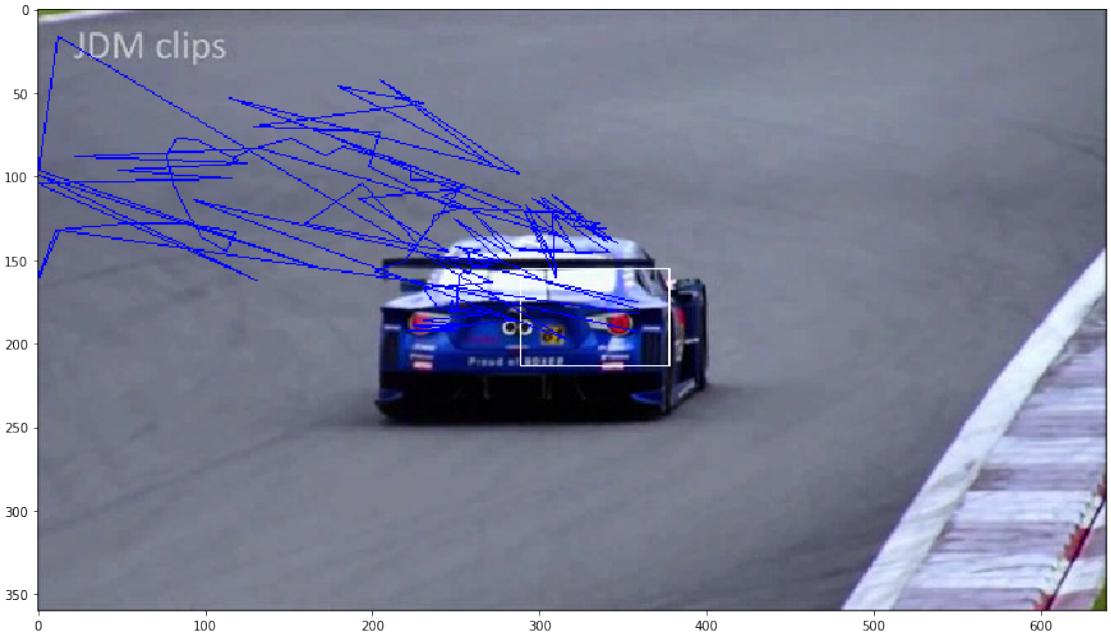

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

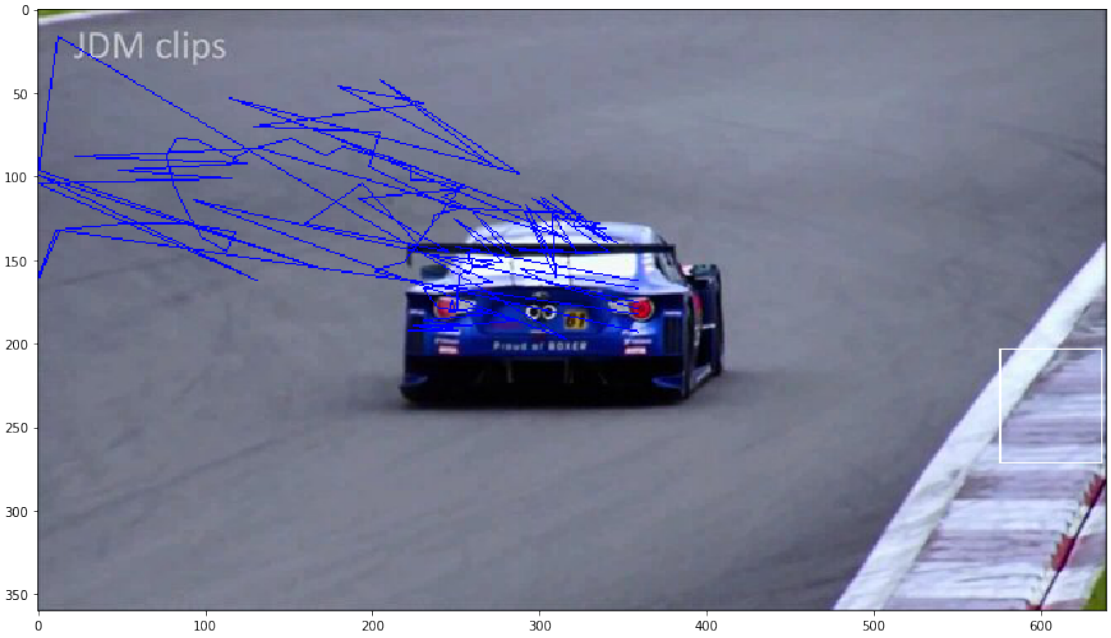

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

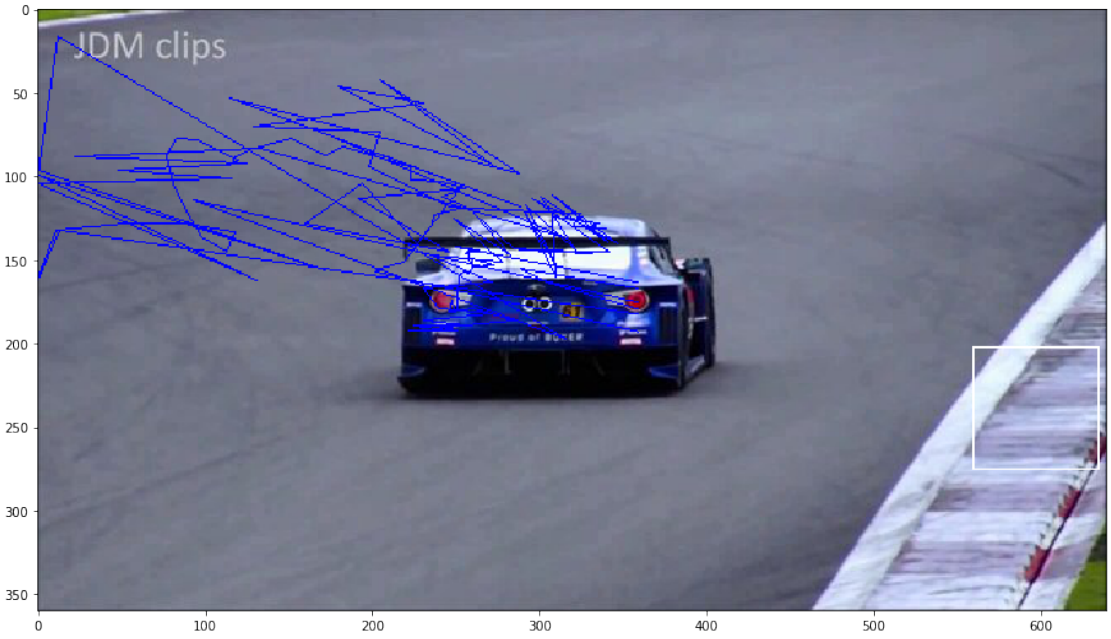

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

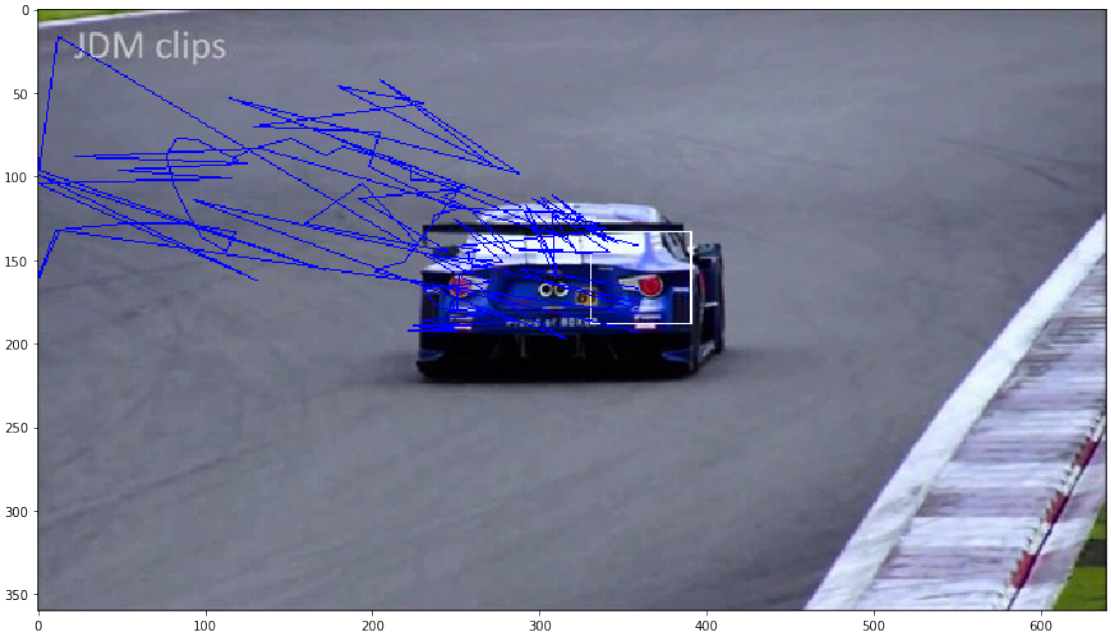

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

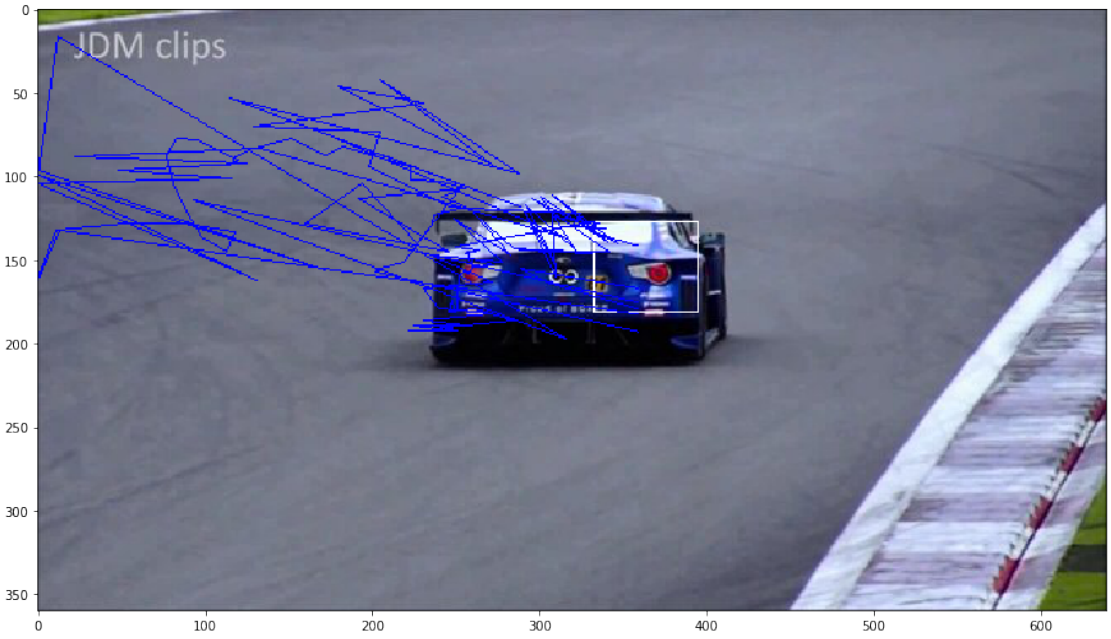

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

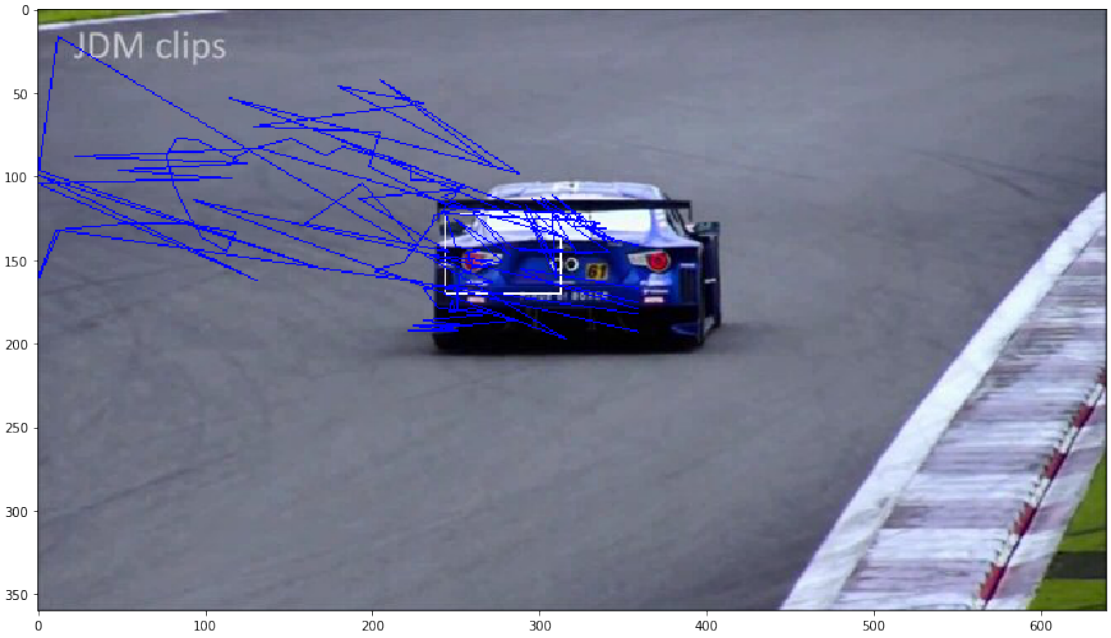

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

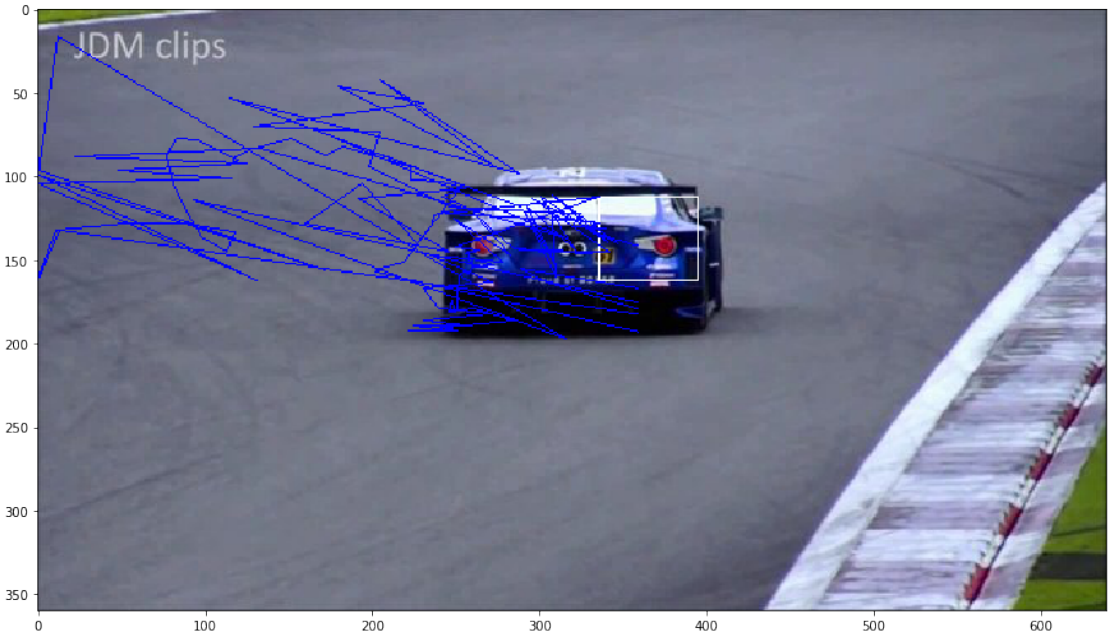

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

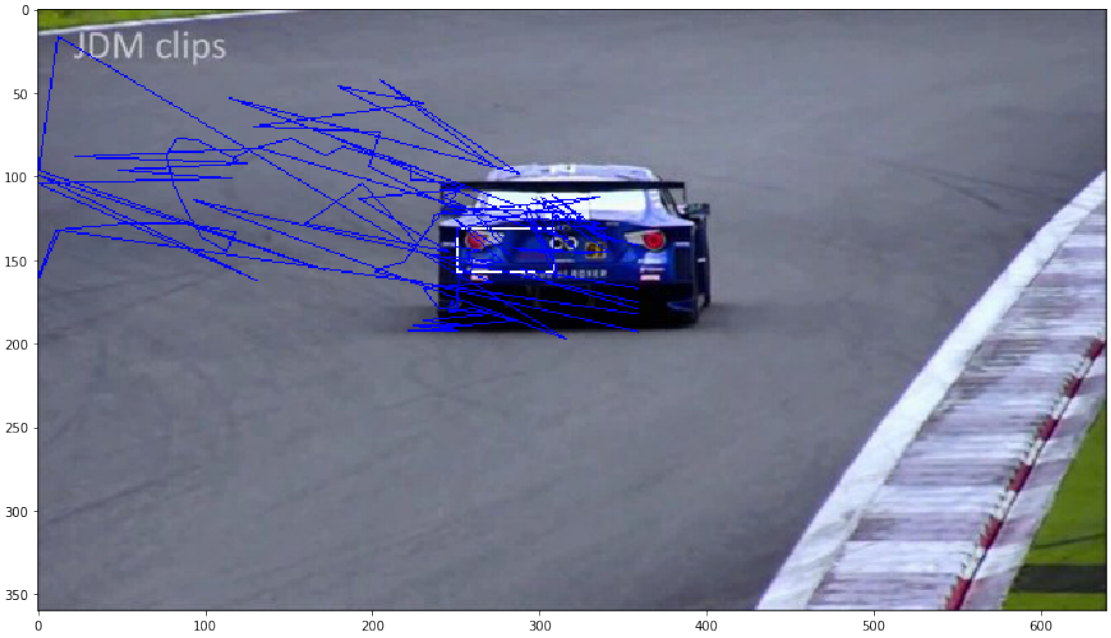

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

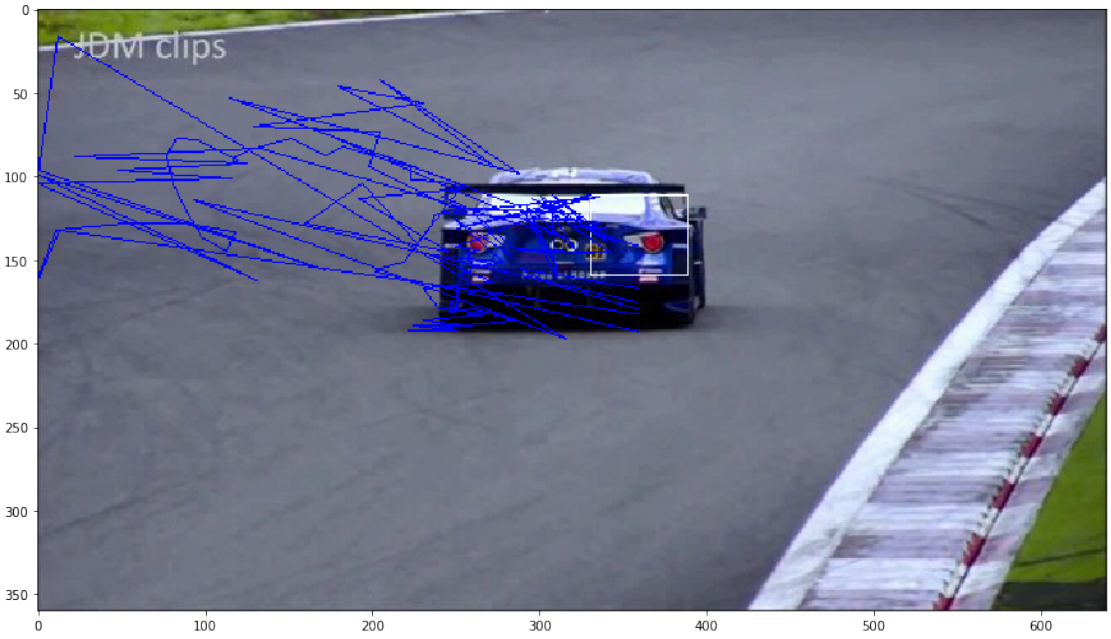

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

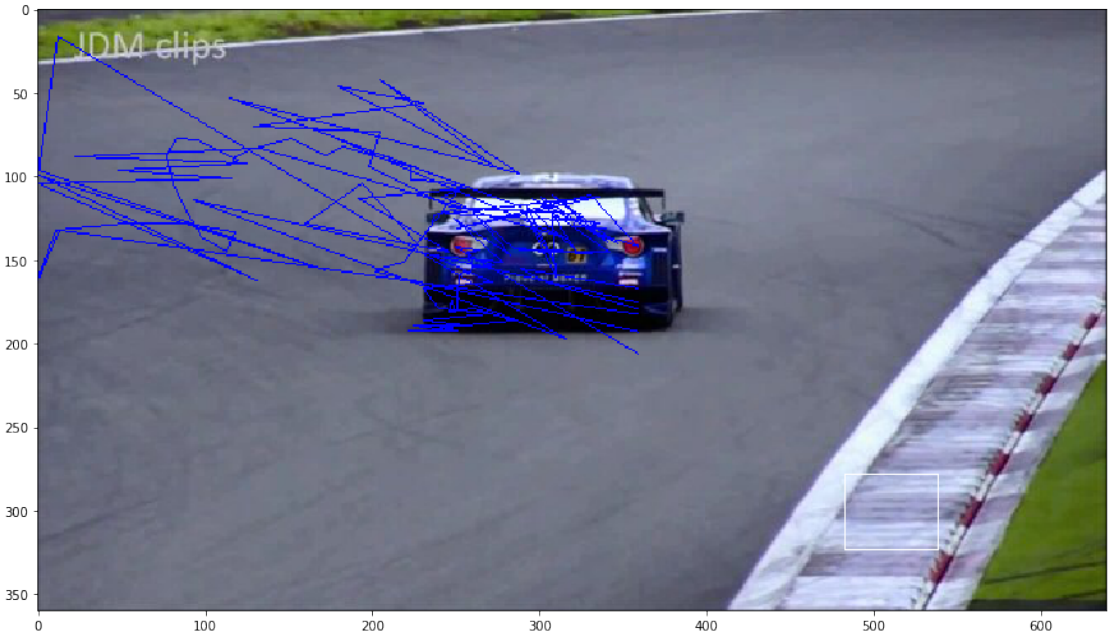

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

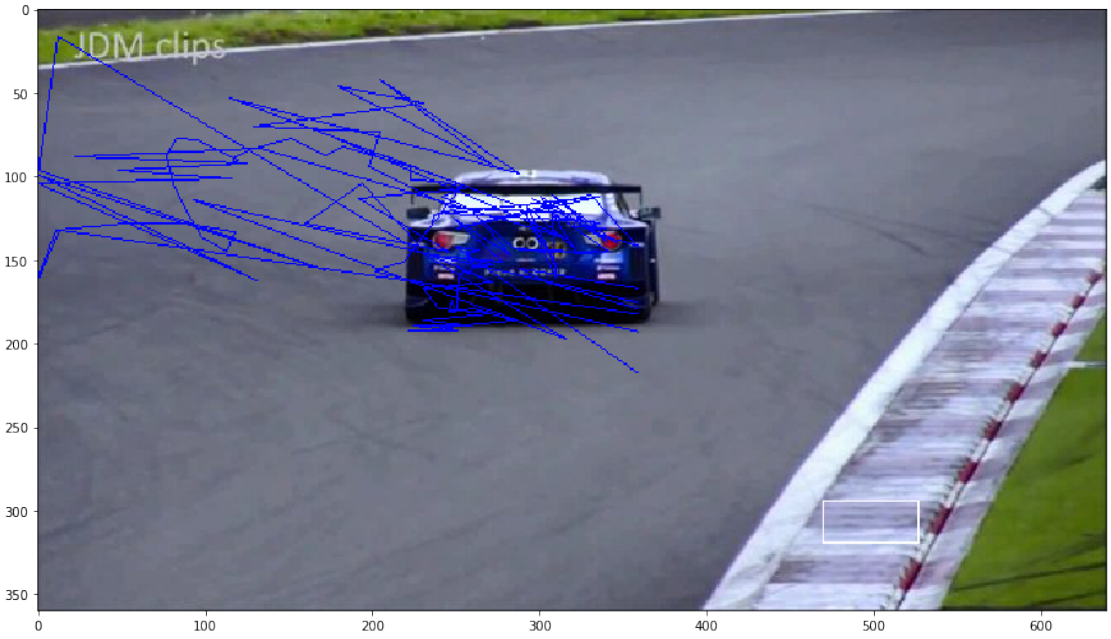

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

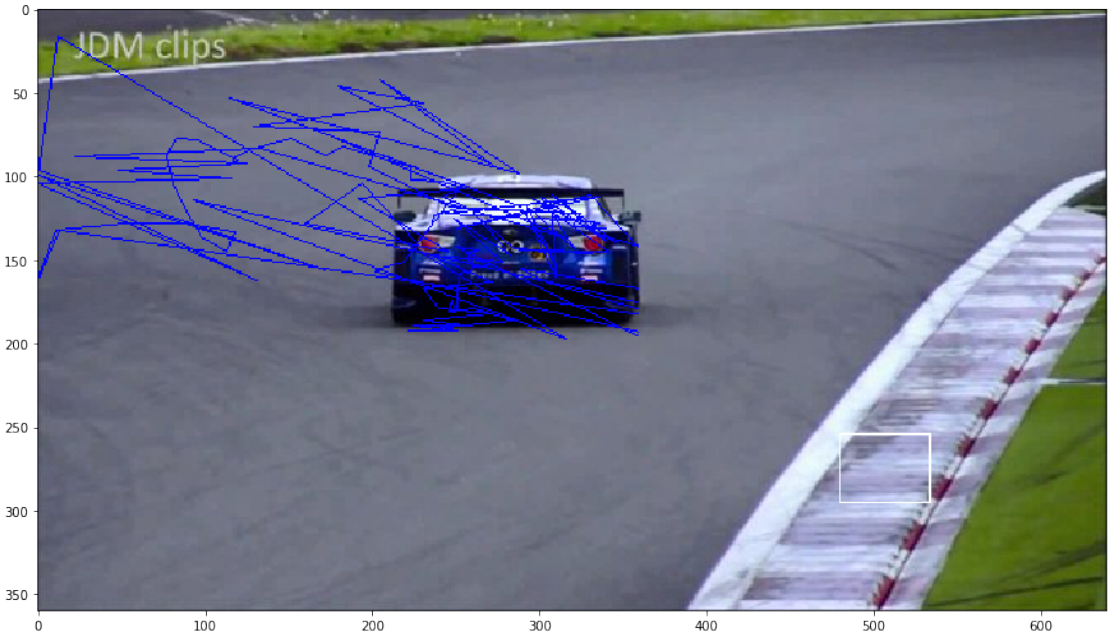

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

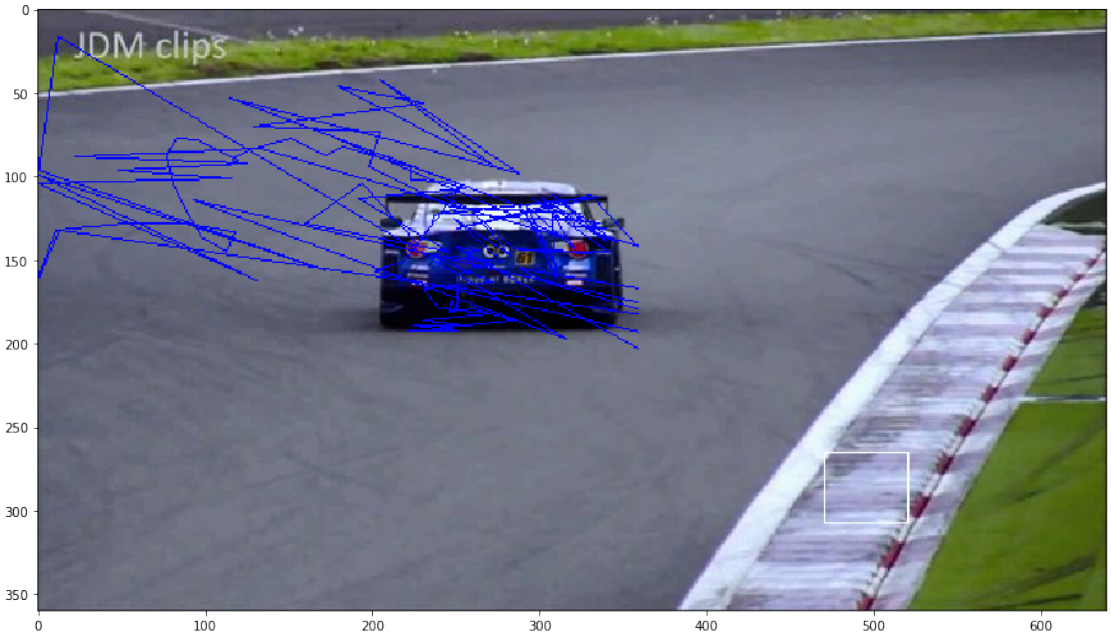

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600

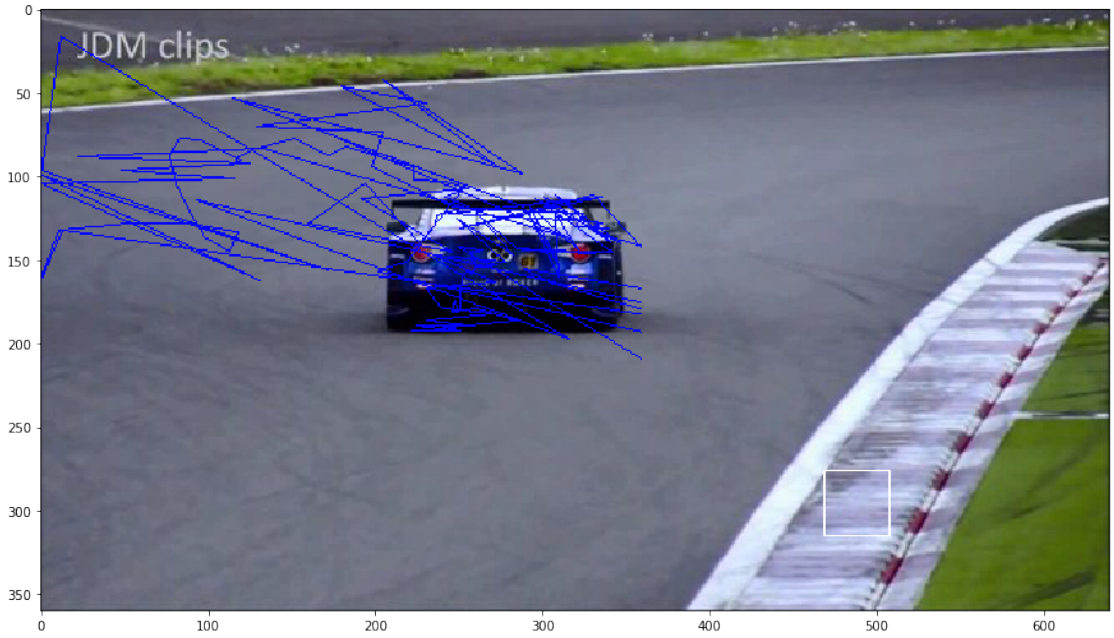

JDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600


IDM clips
0
50
100
150
200
250
300
350
0
100
200
300
400
500
600



In [105]:

```
[(308, 149), (250, 135), (250, 135), (250, 135), (308, 149), (250, 135), (308, 149), (250, 135)
```

In [111]:

```
Out[111]: [[308, 250, 250, 250, 308, 250, 308, 250],
           [149, 135, 135, 135, 149, 135, 149, 135]]
```