

Детекция заведующего
лабораторией робототехники
Дмитрия Дмитриевича по его
толстовке

Проблема

- Дисциплина сотрудника
- Находится ли Дмитрий на рабочем месте?
- Пора приходить в кружок, если Дмитрий На месте
- Оповещение учеников о том, что Дмитрий в лабе

Актуальность

- Дмитрий может выходить из лаборатории, её закрывать, уходить на ужин
- Дмитрий иногда приходит раньше и лаборатория соответственно раньше открывается, а туда можно приходить и доделывать общие проекты.
- Для повышения эффективности работы в лаборатории и оповещения учеников о том, что Дмитрий их ждёт и уже работает



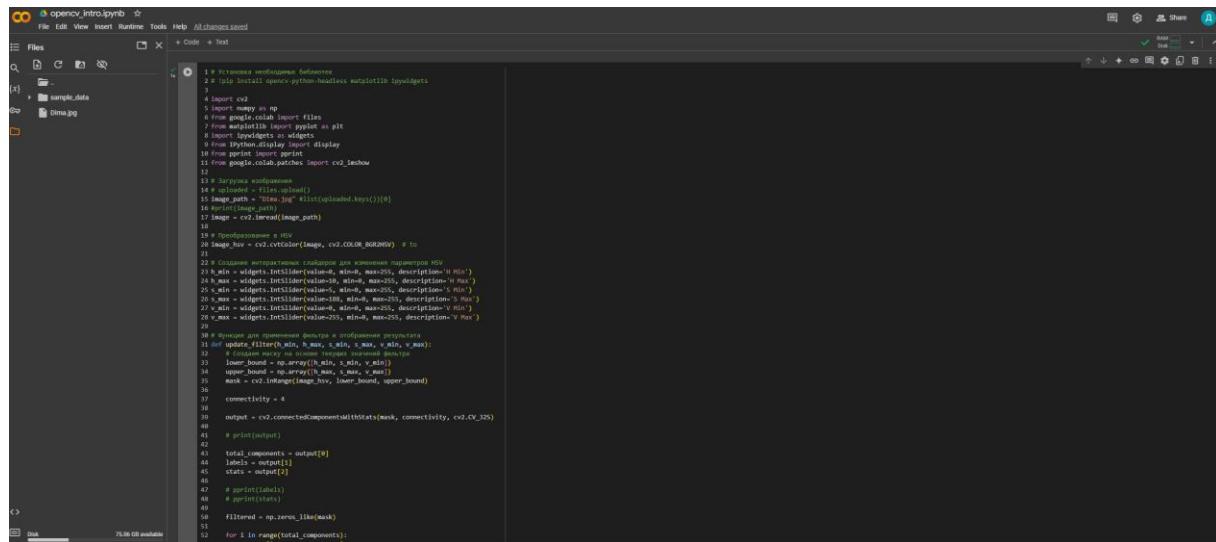
Цель: Детектировать Дмитрия

Задачи:

- 1) Получить изображение
- 2) Преобразовать его в HSV формат
- 3) Настроить маску для детекции только определенного цвета
- 4) Выделить связные компоненты
- 5) Отфильтровать по площади

Продукт

- Скрипт на питоне в google colab, определяющий по фотографии есть ли там Дмитрий в красной толстовке



The screenshot shows a Google Colab notebook titled 'opencv_intro.ipynb'. The code cell contains a Python script for face detection and tracking. It imports necessary libraries like cv2, numpy, and opencv modules. It loads an image from 'Dima.jpg', converts it to HSV color space, applies a Gaussian blur, and creates a mask for the red hoodie. The script then performs connected component analysis on the mask to identify the number of components. Finally, it prints the count of detected faces and displays the original image with a green bounding box around the detected face.

```
1 # Извините за долгое ожидание
2 # pip install opencv-python-headless matplotlib ipywidgets
3
4 import cv2
5 import numpy as np
6 from os import listdir
7 from os.path import join
8 from matplotlib import pyplot as plt
9 import ipywidgets as widgets
10 from IPython.display import display
11 from PIL import Image
12 from google.colab.patches import cv2_imshow
13
14 # Инициализация
15 # !pip install opencv-python
16 image_path = "Dima.jpg" #list(uploaded.files())[0]
17 #print(image_path)
18 image = cv2.imread(image_path)
19
20 # Преобразование в HSV
21 image_hsv = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
22 # Гауссово размытие из-за высокой яркости
23 s_min = widgets.IntSlider(value=0, min=0, max=255, description='H Min')
24 h_max = widgets.IntSlider(value=0, min=0, max=255, description='H Max')
25 s_min = widgets.IntSlider(value=0, min=0, max=255, description='S Min')
26 s_max = widgets.IntSlider(value=255, min=0, max=255, description='S Max')
27 v_min = widgets.IntSlider(value=0, min=0, max=255, description='V Min')
28 v_max = widgets.IntSlider(value=255, min=0, max=255, description='V Max')
29
30 # Дискреционное изменение порога
31 def update_filter(s_min, h_max, s_max, v_min, v_max):
32     lower_bound = np.array([h_min, s_min, v_min])
33     upper_bound = np.array([h_max, s_max, v_max])
34     mask = cv2.inRange(image_hsv, lower_bound, upper_bound)
35
36 connectivity = 4
37 output = cv2.connectedComponentsWithStats(mask, connectivity, cv2.CV_32S)
38
39 # print(output)
40 total_components = output[0]
41 labels = output[1]
42 stats = output[2]
43
44 # print(labels)
45 # print(stats)
46
47 filtered = np.zeros_like(mask)
48
49 for i in range(total_components):
50     if stats[i][4] > 1000000:
51         filtered[labels == i] = 255
52
53 filtered = np.zeros_like(mask)
54
55 for i in range(total_components):
56     if stats[i][4] > 1000000:
57         filtered[labels == i] = 255
58
59 filtered = np.zeros_like(mask)
60
61 for i in range(total_components):
62     if stats[i][4] > 1000000:
63         filtered[labels == i] = 255
64
65 filtered = np.zeros_like(mask)
66
67 for i in range(total_components):
68     if stats[i][4] > 1000000:
69         filtered[labels == i] = 255
70
71 filtered = np.zeros_like(mask)
72
73 for i in range(total_components):
74     if stats[i][4] > 1000000:
75         filtered[labels == i] = 255
76
77 filtered = np.zeros_like(mask)
78
79 for i in range(total_components):
80     if stats[i][4] > 1000000:
81         filtered[labels == i] = 255
82
83 filtered = np.zeros_like(mask)
84
85 for i in range(total_components):
86     if stats[i][4] > 1000000:
87         filtered[labels == i] = 255
88
89 filtered = np.zeros_like(mask)
90
91 for i in range(total_components):
92     if stats[i][4] > 1000000:
93         filtered[labels == i] = 255
94
95 filtered = np.zeros_like(mask)
96
97 for i in range(total_components):
98     if stats[i][4] > 1000000:
99         filtered[labels == i] = 255
100
101 filtered = np.zeros_like(mask)
102
103 for i in range(total_components):
104     if stats[i][4] > 1000000:
105         filtered[labels == i] = 255
106
107 filtered = np.zeros_like(mask)
108
109 for i in range(total_components):
110     if stats[i][4] > 1000000:
111         filtered[labels == i] = 255
112
113 filtered = np.zeros_like(mask)
114
115 for i in range(total_components):
116     if stats[i][4] > 1000000:
117         filtered[labels == i] = 255
118
119 filtered = np.zeros_like(mask)
120
121 for i in range(total_components):
122     if stats[i][4] > 1000000:
123         filtered[labels == i] = 255
124
125 filtered = np.zeros_like(mask)
126
127 for i in range(total_components):
128     if stats[i][4] > 1000000:
129         filtered[labels == i] = 255
130
131 filtered = np.zeros_like(mask)
132
133 for i in range(total_components):
134     if stats[i][4] > 1000000:
135         filtered[labels == i] = 255
136
137 filtered = np.zeros_like(mask)
138
139 for i in range(total_components):
140     if stats[i][4] > 1000000:
141         filtered[labels == i] = 255
142
143 filtered = np.zeros_like(mask)
144
145 for i in range(total_components):
146     if stats[i][4] > 1000000:
147         filtered[labels == i] = 255
148
149 filtered = np.zeros_like(mask)
150
151 for i in range(total_components):
152     if stats[i][4] > 1000000:
153         filtered[labels == i] = 255
154
155 filtered = np.zeros_like(mask)
156
157 for i in range(total_components):
158     if stats[i][4] > 1000000:
159         filtered[labels == i] = 255
160
161 filtered = np.zeros_like(mask)
162
163 for i in range(total_components):
164     if stats[i][4] > 1000000:
165         filtered[labels == i] = 255
166
167 filtered = np.zeros_like(mask)
168
169 for i in range(total_components):
170     if stats[i][4] > 1000000:
171         filtered[labels == i] = 255
172
173 filtered = np.zeros_like(mask)
174
175 for i in range(total_components):
176     if stats[i][4] > 1000000:
177         filtered[labels == i] = 255
178
179 filtered = np.zeros_like(mask)
180
181 for i in range(total_components):
182     if stats[i][4] > 1000000:
183         filtered[labels == i] = 255
184
185 filtered = np.zeros_like(mask)
186
187 for i in range(total_components):
188     if stats[i][4] > 1000000:
189         filtered[labels == i] = 255
190
191 filtered = np.zeros_like(mask)
192
193 for i in range(total_components):
194     if stats[i][4] > 1000000:
195         filtered[labels == i] = 255
196
197 filtered = np.zeros_like(mask)
198
199 for i in range(total_components):
200     if stats[i][4] > 1000000:
201         filtered[labels == i] = 255
202
203 filtered = np.zeros_like(mask)
204
205 for i in range(total_components):
206     if stats[i][4] > 1000000:
207         filtered[labels == i] = 255
208
209 filtered = np.zeros_like(mask)
210
211 for i in range(total_components):
212     if stats[i][4] > 1000000:
213         filtered[labels == i] = 255
214
215 filtered = np.zeros_like(mask)
216
217 for i in range(total_components):
218     if stats[i][4] > 1000000:
219         filtered[labels == i] = 255
220
221 filtered = np.zeros_like(mask)
222
223 for i in range(total_components):
224     if stats[i][4] > 1000000:
225         filtered[labels == i] = 255
226
227 filtered = np.zeros_like(mask)
228
229 for i in range(total_components):
230     if stats[i][4] > 1000000:
231         filtered[labels == i] = 255
232
233 filtered = np.zeros_like(mask)
234
235 for i in range(total_components):
236     if stats[i][4] > 1000000:
237         filtered[labels == i] = 255
238
239 filtered = np.zeros_like(mask)
240
241 for i in range(total_components):
242     if stats[i][4] > 1000000:
243         filtered[labels == i] = 255
244
245 filtered = np.zeros_like(mask)
246
247 for i in range(total_components):
248     if stats[i][4] > 1000000:
249         filtered[labels == i] = 255
250
251 filtered = np.zeros_like(mask)
252
253 for i in range(total_components):
254     if stats[i][4] > 1000000:
255         filtered[labels == i] = 255
256
257 filtered = np.zeros_like(mask)
258
259 for i in range(total_components):
260     if stats[i][4] > 1000000:
261         filtered[labels == i] = 255
262
263 filtered = np.zeros_like(mask)
264
265 for i in range(total_components):
266     if stats[i][4] > 1000000:
267         filtered[labels == i] = 255
268
269 filtered = np.zeros_like(mask)
270
271 for i in range(total_components):
272     if stats[i][4] > 1000000:
273         filtered[labels == i] = 255
274
275 filtered = np.zeros_like(mask)
276
277 for i in range(total_components):
278     if stats[i][4] > 1000000:
279         filtered[labels == i] = 255
280
281 filtered = np.zeros_like(mask)
282
283 for i in range(total_components):
284     if stats[i][4] > 1000000:
285         filtered[labels == i] = 255
286
287 filtered = np.zeros_like(mask)
288
289 for i in range(total_components):
290     if stats[i][4] > 1000000:
291         filtered[labels == i] = 255
292
293 filtered = np.zeros_like(mask)
294
295 for i in range(total_components):
296     if stats[i][4] > 1000000:
297         filtered[labels == i] = 255
298
299 filtered = np.zeros_like(mask)
300
301 for i in range(total_components):
302     if stats[i][4] > 1000000:
303         filtered[labels == i] = 255
304
305 filtered = np.zeros_like(mask)
306
307 for i in range(total_components):
308     if stats[i][4] > 1000000:
309         filtered[labels == i] = 255
310
311 filtered = np.zeros_like(mask)
312
313 for i in range(total_components):
314     if stats[i][4] > 1000000:
315         filtered[labels == i] = 255
316
317 filtered = np.zeros_like(mask)
318
319 for i in range(total_components):
320     if stats[i][4] > 1000000:
321         filtered[labels == i] = 255
322
323 filtered = np.zeros_like(mask)
324
325 for i in range(total_components):
326     if stats[i][4] > 1000000:
327         filtered[labels == i] = 255
328
329 filtered = np.zeros_like(mask)
330
330 for i in range(total_components):
331     if stats[i][4] > 1000000:
332         filtered[labels == i] = 255
333
334 filtered = np.zeros_like(mask)
335
335 for i in range(total_components):
336     if stats[i][4] > 1000000:
337         filtered[labels == i] = 255
338
339 filtered = np.zeros_like(mask)
340
340 for i in range(total_components):
341     if stats[i][4] > 1000000:
342         filtered[labels == i] = 255
343
344 filtered = np.zeros_like(mask)
345
345 for i in range(total_components):
346     if stats[i][4] > 1000000:
347         filtered[labels == i] = 255
348
349 filtered = np.zeros_like(mask)
350
350 for i in range(total_components):
351     if stats[i][4] > 1000000:
352         filtered[labels == i] = 255
353
354 filtered = np.zeros_like(mask)
355
355 for i in range(total_components):
356     if stats[i][4] > 1000000:
357         filtered[labels == i] = 255
358
359 filtered = np.zeros_like(mask)
360
360 for i in range(total_components):
361     if stats[i][4] > 1000000:
362         filtered[labels == i] = 255
363
364 filtered = np.zeros_like(mask)
365
365 for i in range(total_components):
366     if stats[i][4] > 1000000:
367         filtered[labels == i] = 255
368
369 filtered = np.zeros_like(mask)
370
370 for i in range(total_components):
371     if stats[i][4] > 1000000:
372         filtered[labels == i] = 255
373
374 filtered = np.zeros_like(mask)
375
375 for i in range(total_components):
376     if stats[i][4] > 1000000:
377         filtered[labels == i] = 255
378
379 filtered = np.zeros_like(mask)
380
380 for i in range(total_components):
381     if stats[i][4] > 1000000:
382         filtered[labels == i] = 255
383
384 filtered = np.zeros_like(mask)
385
385 for i in range(total_components):
386     if stats[i][4] > 1000000:
387         filtered[labels == i] = 255
388
389 filtered = np.zeros_like(mask)
390
390 for i in range(total_components):
391     if stats[i][4] > 1000000:
392         filtered[labels == i] = 255
393
394 filtered = np.zeros_like(mask)
395
395 for i in range(total_components):
396     if stats[i][4] > 1000000:
397         filtered[labels == i] = 255
398
399 filtered = np.zeros_like(mask)
400
400 for i in range(total_components):
401     if stats[i][4] > 1000000:
402         filtered[labels == i] = 255
403
404 filtered = np.zeros_like(mask)
405
405 for i in range(total_components):
406     if stats[i][4] > 1000000:
407         filtered[labels == i] = 255
408
409 filtered = np.zeros_like(mask)
410
410 for i in range(total_components):
411     if stats[i][4] > 1000000:
412         filtered[labels == i] = 255
413
414 filtered = np.zeros_like(mask)
415
415 for i in range(total_components):
416     if stats[i][4] > 1000000:
417         filtered[labels == i] = 255
418
419 filtered = np.zeros_like(mask)
420
420 for i in range(total_components):
421     if stats[i][4] > 1000000:
422         filtered[labels == i] = 255
423
424 filtered = np.zeros_like(mask)
425
425 for i in range(total_components):
426     if stats[i][4] > 1000000:
427         filtered[labels == i] = 255
428
429 filtered = np.zeros_like(mask)
430
430 for i in range(total_components):
431     if stats[i][4] > 1000000:
432         filtered[labels == i] = 255
433
434 filtered = np.zeros_like(mask)
435
435 for i in range(total_components):
436     if stats[i][4] > 1000000:
437         filtered[labels == i] = 255
438
439 filtered = np.zeros_like(mask)
440
440 for i in range(total_components):
441     if stats[i][4] > 1000000:
442         filtered[labels == i] = 255
443
444 filtered = np.zeros_like(mask)
445
445 for i in range(total_components):
446     if stats[i][4] > 1000000:
447         filtered[labels == i] = 255
448
449 filtered = np.zeros_like(mask)
450
450 for i in range(total_components):
451     if stats[i][4] > 1000000:
452         filtered[labels == i] = 255
453
454 filtered = np.zeros_like(mask)
455
455 for i in range(total_components):
456     if stats[i][4] > 1000000:
457         filtered[labels == i] = 255
458
459 filtered = np.zeros_like(mask)
460
460 for i in range(total_components):
461     if stats[i][4] > 1000000:
462         filtered[labels == i] = 255
463
464 filtered = np.zeros_like(mask)
465
465 for i in range(total_components):
466     if stats[i][4] > 1000000:
467         filtered[labels == i] = 255
468
469 filtered = np.zeros_like(mask)
470
470 for i in range(total_components):
471     if stats[i][4] > 1000000:
472         filtered[labels == i] = 255
473
474 filtered = np.zeros_like(mask)
475
475 for i in range(total_components):
476     if stats[i][4] > 1000000:
477         filtered[labels == i] = 255
478
479 filtered = np.zeros_like(mask)
480
480 for i in range(total_components):
481     if stats[i][4] > 1000000:
482         filtered[labels == i] = 255
483
484 filtered = np.zeros_like(mask)
485
485 for i in range(total_components):
486     if stats[i][4] > 1000000:
487         filtered[labels == i] = 255
488
489 filtered = np.zeros_like(mask)
490
490 for i in range(total_components):
491     if stats[i][4] > 1000000:
492         filtered[labels == i] = 255
493
494 filtered = np.zeros_like(mask)
495
495 for i in range(total_components):
496     if stats[i][4] > 1000000:
497         filtered[labels == i] = 255
498
499 filtered = np.zeros_like(mask)
500
500 for i in range(total_components):
501     if stats[i][4] > 1000000:
502         filtered[labels == i] = 255
503
504 filtered = np.zeros_like(mask)
505
505 for i in range(total_components):
506     if stats[i][4] > 1000000:
507         filtered[labels == i] = 255
508
509 filtered = np.zeros_like(mask)
510
510 for i in range(total_components):
511     if stats[i][4] > 1000000:
512         filtered[labels == i] = 255
513
514 filtered = np.zeros_like(mask)
515
515 for i in range(total_components):
516     if stats[i][4] > 1000000:
517         filtered[labels == i] = 255
518
519 filtered = np.zeros_like(mask)
520
520 for i in range(total_components):
521     if stats[i][4] > 1000000:
522         filtered[labels == i] = 255
523
524 filtered = np.zeros_like(mask)
525
525 for i in range(total_components):
526     if stats[i][4] > 1000000:
527         filtered[labels == i] = 255
528
529 filtered = np.zeros_like(mask)
530
530 for i in range(total_components):
531     if stats[i][4] > 1000000:
532         filtered[labels == i] = 255
533
534 filtered = np.zeros_like(mask)
535
535 for i in range(total_components):
536     if stats[i][4] > 1000000:
537         filtered[labels == i] = 255
538
539 filtered = np.zeros_like(mask)
540
540 for i in range(total_components):
541     if stats[i][4] > 1000000:
542         filtered[labels == i] = 255
543
544 filtered = np.zeros_like(mask)
545
545 for i in range(total_components):
546     if stats[i][4] > 1000000:
547         filtered[labels == i] = 255
548
549 filtered = np.zeros_like(mask)
550
550 for i in range(total_components):
551     if stats[i][4] > 1000000:
552         filtered[labels == i] = 255
553
554 filtered = np.zeros_like(mask)
555
555 for i in range(total_components):
556     if stats[i][4] > 1000000:
557         filtered[labels == i] = 255
558
559 filtered = np.zeros_like(mask)
560
560 for i in range(total_components):
561     if stats[i][4] > 1000000:
562         filtered[labels == i] = 255
563
564 filtered = np.zeros_like(mask)
565
565 for i in range(total_components):
566     if stats[i][4] > 1000000:
567         filtered[labels == i] = 255
568
569 filtered = np.zeros_like(mask)
570
570 for i in range(total_components):
571     if stats[i][4] > 1000000:
572         filtered[labels == i] = 255
573
574 filtered = np.zeros_like(mask)
575
575 for i in range(total_components):
576     if stats[i][4] > 1000000:
577         filtered[labels == i] = 255
578
579 filtered = np.zeros_like(mask)
580
580 for i in range(total_components):
581     if stats[i][4] > 1000000:
582         filtered[labels == i] = 255
583
584 filtered = np.zeros_like(mask)
585
585 for i in range(total_components):
586     if stats[i][4] > 1000000:
587         filtered[labels == i] = 255
588
589 filtered = np.zeros_like(mask)
590
590 for i in range(total_components):
591     if stats[i][4] > 1000000:
592         filtered[labels == i] = 255
593
594 filtered = np.zeros_like(mask)
595
595 for i in range(total_components):
596     if stats[i][4] > 1000000:
597         filtered[labels == i] = 255
598
599 filtered = np.zeros_like(mask)
600
600 for i in range(total_components):
601     if stats[i][4] > 1000000:
602         filtered[labels == i] = 255
603
604 filtered = np.zeros_like(mask)
605
605 for i in range(total_components):
606     if stats[i][4] > 1000000:
607         filtered[labels == i] = 255
608
609 filtered = np.zeros_like(mask)
610
610 for i in range(total_components):
611     if stats[i][4] > 1000000:
612         filtered[labels == i] = 255
613
614 filtered = np.zeros_like(mask)
615
615 for i in range(total_components):
616     if stats[i][4] > 1000000:
617         filtered[labels == i] = 255
618
619 filtered = np.zeros_like(mask)
620
620 for i in range(total_components):
621     if stats[i][4] > 1000000:
622         filtered[labels == i] = 255
623
624 filtered = np.zeros_like(mask)
625
625 for i in range(total_components):
626     if stats[i][4] > 1000000:
627         filtered[labels == i] = 255
628
629 filtered = np.zeros_like(mask)
630
630 for i in range(total_components):
631     if stats[i][4] > 1000000:
632         filtered[labels == i] = 255
633
634 filtered = np.zeros_like(mask)
635
635 for i in range(total_components):
636     if stats[i][4] > 1000000:
637         filtered[labels == i] = 255
638
639 filtered = np.zeros_like(mask)
640
640 for i in range(total_components):
641     if stats[i][4] > 1000000:
642         filtered[labels == i] = 255
643
644 filtered = np.zeros_like(mask)
645
645 for i in range(total_components):
646     if stats[i][4] > 1000000:
647         filtered[labels == i] = 255
648
649 filtered = np.zeros_like(mask)
650
650 for i in range(total_components):
651     if stats[i][4] > 1000000:
652         filtered[labels == i] = 255
653
654 filtered = np.zeros_like(mask)
655
655 for i in range(total_components):
656     if stats[i][4] > 1000000:
657         filtered[labels == i] = 255
658
659 filtered = np.zeros_like(mask)
660
660 for i in range(total_components):
661     if stats[i][4] > 1000000:
662         filtered[labels == i] = 255
663
664 filtered = np.zeros_like(mask)
665
665 for i in range(total_components):
666     if stats[i][4] > 1000000:
667         filtered[labels == i] = 255
668
669 filtered = np.zeros_like(mask)
670
670 for i in range(total_components):
671     if stats[i][4] > 1000000:
672         filtered[labels == i] = 255
673
674 filtered = np.zeros_like(mask)
675
675 for i in range(total_components):
676     if stats[i][4] > 1000000:
677         filtered[labels == i] = 255
678
679 filtered = np.zeros_like(mask)
680
680 for i in range(total_components):
681     if stats[i][4] > 1000000:
682         filtered[labels == i] = 255
683
684 filtered = np.zeros_like(mask)
685
685 for i in range(total_components):
686     if stats[i][4] > 1000000:
687         filtered[labels == i] = 255
688
689 filtered = np.zeros_like(mask)
690
690 for i in range(total_components):
691     if stats[i][4] > 1000000:
692         filtered[labels == i] = 255
693
694 filtered = np.zeros_like(mask)
695
695 for i in range(total_components):
696     if stats[i][4] > 1000000:
697         filtered[labels == i] = 255
698
699 filtered = np.zeros_like(mask)
700
700 for i in range(total_components):
701     if stats[i][4] > 1000000:
702         filtered[labels == i] = 255
703
704 filtered = np.zeros_like(mask)
705
705 for i in range(total_components):
706     if stats[i][4] > 1000000:
707         filtered[labels == i] = 255
708
709 filtered = np.zeros_like(mask)
710
710 for i in range(total_components):
711     if stats[i][4] > 1000000:
712         filtered[labels == i] = 255
713
714 filtered = np.zeros_like(mask)
715
715 for i in range(total_components):
716     if stats[i][4] > 1000000:
717         filtered[labels == i] = 255
718
719 filtered = np.zeros_like(mask)
720
720 for i in range(total_components):
721     if stats[i][4] > 1000000:
722         filtered[labels == i] = 255
723
724 filtered = np.zeros_like(mask)
725
725 for i in range(total_components):
726     if stats[i][4] > 1000000:
727         filtered[labels == i] = 255
728
729 filtered = np.zeros_like(mask)
730
730 for i in range(total_components):
731     if stats[i][4] > 1000000:
732         filtered[labels == i] = 255
733
734 filtered = np.zeros_like(mask)
735
735 for i in range(total_components):
736     if stats[i][4] > 1000000:
737         filtered[labels == i] = 255
738
739 filtered = np.zeros_like(mask)
740
740 for i in range(total_components):
741     if stats[i][4] > 1000000:
742         filtered[labels == i] = 255
743
744 filtered = np.zeros_like(mask)
745
745 for i in range(total_components):
746     if stats[i][4] > 1000000:
747         filtered[labels == i] = 255
748
749 filtered = np.zeros_like(mask)
750
750 for i in range(total_components):
751     if stats[i][4] > 1000000:
752         filtered[labels == i] = 255
753
754 filtered = np.zeros_like(mask)
755
755 for i in range(total_components):
756     if stats[i][4] > 1000000:
757         filtered[labels == i] = 255
758
759 filtered = np.zeros_like(mask)
760
760 for i in range(total_components):
761     if stats[i][4] > 1000000:
762         filtered[labels == i] = 255
763
764 filtered = np.zeros_like(mask)
765
765 for i in range(total_components):
766     if stats[i][4] > 1000000:
767         filtered[labels == i] = 255
768
769 filtered = np.zeros_like(mask)
770
770 for i in range(total_components):
771     if stats[i][4] > 1000000:
772         filtered[labels == i] = 255
773
774 filtered = np.zeros_like(mask)
775
775 for i in range(total_components):
776     if stats[i][4] > 1000000:
777         filtered[labels == i] = 255
778
779 filtered = np.zeros_like(mask)
780
780 for i in range(total_components):
781     if stats[i][4] > 1000000:
782         filtered[labels == i] = 255
783
784 filtered = np.zeros_like(mask)
785
785 for i in range(total_components):
786     if stats[i][4] > 1000000:
787         filtered[labels == i] = 255
788
789 filtered = np.zeros_like(mask)
790
790 for i in range(total_components):
791     if stats[i][4] > 1000000:
792         filtered[labels == i] = 255
793
794 filtered = np.zeros_like(mask)
795
795 for i in range(total_components):
796     if stats[i][4] > 1000000:
797         filtered[labels == i] = 255
798
799 filtered = np.zeros_like(mask)
800
800 for i in range(total_components):
801     if stats[i][4] > 1000000:
802         filtered[labels == i] = 255
803
804 filtered = np.zeros_like(mask)
805
805 for i in range(total_components):
806     if stats[i][4] > 1000000:
807         filtered[labels == i] = 255
808
809 filtered = np.zeros_like(mask)
810
810 for i in range(total_components):
811     if stats[i][4] > 1000000:
812         filtered[labels == i] = 255
813
814 filtered = np.zeros_like(mask)
815
815 for i in range(total_components):
816     if stats[i][4] > 1000000:
817         filtered[labels == i] = 255
818
819 filtered = np.zeros_like(mask)
820
820 for i in range(total_components):
821     if stats[i][4] > 1000000:
822         filtered[labels == i] = 255
823
824 filtered = np.zeros_like(mask)
825
825 for i in range(total_components):
826     if stats[i][4] > 1000000:
827         filtered[labels == i] = 255
828
829 filtered = np.zeros_like(mask)
830
830 for i in range(total_components):
831     if stats[i][4] > 1000000:
832         filtered[labels == i] = 255
833
834 filtered = np.zeros_like(mask)
835
835 for i in range(total_components):
836     if stats[i][4] > 1000000:
837         filtered[labels == i] = 255
838
839 filtered = np.zeros_like(mask)
840
840 for i in range(total_components):
841     if stats[i][4] > 1000000:
842         filtered[labels == i] = 255
843
844 filtered = np.zeros_like(mask)
845
845 for i in range(total_components):
846     if stats[i][4] > 1000000:
847         filtered[labels == i] = 255
848
849 filtered = np.zeros_like(mask)
850
850 for i in range(total_components):
851     if stats[i][4] > 1000000:
852         filtered[labels == i] = 255
853
854 filtered = np.zeros_like(mask)
855
855 for i in range(total_components):
856     if stats[i][4] > 1000000:
857         filtered[labels == i] = 255
858
859 filtered = np.zeros_like(mask)
860
860 for i in range(total_components):
861     if stats[i][4] > 1000000:
862         filtered[labels == i] = 255
863
864 filtered = np.zeros_like(mask)
865
865 for i in range(total_components):
866     if stats[i][4] > 1000000:
867         filtered[labels == i] = 255
868
869 filtered = np.zeros_like(mask)
870
870 for i in range(total_components):
871     if stats[i][4] > 1000000:
872         filtered[labels == i] = 255
873
874 filtered = np.zeros
```

Конкуренты, аналоги

Конкуренты (обосновать):

- 1) Охранные системы
- 2) Google
- 3) Илон Маск и SpaceX
- 4) Нейросети YOLO

Новизна

Абсолютно инновационный project в сфере технического зрения с фильтром по площади объекта

Перспективы развития

Расширенная слежка за Дмитрием по всему технопарку с камер охранной системы.

Проблема может быть только, если ещё кто то кроме него будет ходить в красных толстовках.

Ссылки на проект

<https://colab.research.google.com/drive/1Lcp79ve0GLjKeGnYXf2zH6JvuO5eylw#scrollTo= IR72DQs1BGy>

Спасибо за внимание

- Дмитрий Устинов
- <http://t.me/dimaystinov>

Какие темы должны быть

- 1. Название проекта.
- 2. Описание проблемы которую вы хотите решить.
- 3. Актуальность.
- 4. Цели, Задачи.
- 5. Описание продукта (прототип).
- 6. Конкуренты, аналоги, таблица сравнения.
- 7. Новизна.
- 8. Перспективы развития.
- 9. Спасибо за внимание, ваш телефон, почта и др. реквизиты.
- 10. Ссылка на гитхаб (или колаб) и на видео