brutForce (Calls: 1000, Time: 535.417 s)

Generated 28-дек.-2020 05:25:47 using performance time.
Function in file C:\Users\yarba\Desktop\ДО\course_4\code\brutForce.m
Copy to new window for comparing multiple runs

Parents (calling functions)

Function Name	Function Type	Calls
ImageProcessing	Script	1000

Lines that take the most time

Line Number	Code	Calls	Total Time (s)	% Time	Time Plot
34	`if(squeeze(imageGS(j, i, :)) ~= 0)`	208981920	520.624	97.2%	
42	`end`	208981920	6.964	1.3%	
41	`end`	208981920	6.759	1.3%	
35	`coloredX(end + 1) = i;`	1484025	0.370	0.1%	
36	`coloredY(end + 1) = j;`	1484025	0.255	0.0%	
All other lines			0.445	0.1%	
Totals			535.417	100%	

Children (called functions)

Function Name	Function Type	Calls	Total Time (s)	% Time	Time Plot
squeeze	Function	208981920	186.806	34.9%	
deg2rad	Function	1000	0.006	0.0%	
Self time (built-ins, overhead, etc.)			348.605	65.1%	
Totals			535.417	100%	

Code Analyzer results

Line Number	Message
4	The value assigned to variable 't' might be unused.
5	The value assigned to variable 'h' might be unused.
11	The value assigned to variable 'pxH' might be unused.
18	The value assigned to variable 'yPx' might be unused.
19	The value assigned to variable 'defPxRadius' might be unused.
23	The value assigned to variable 'x' might be unused.
24	The value assigned to variable 'y' might be unused.
35	The variable 'coloredX' appears to change size on every loop iteration. Consider preallocat…
36	The variable 'coloredY' appears to change size on every loop iteration. Consider preallocat…
57	Using ISEMPTY is usually faster than comparing LENGTH to 0.
57	Using ISEMPTY is usually faster than comparing LENGTH to 0.

Coverage results

Show coverage for parent folder

Total lines in function	70
Non-code lines (comments, blank lines)	21
Code lines (lines that can run)	49
Code lines that did run	49
Code lines that did not run	0
Coverage (did run/can run)	100.00 %

Function listing

Time	Calls	Line	
		1	`function [brutX, brutY] = brutForce(imageGS)`
		2	`% параметры устройства`
< 0.001	1000	3	`d = 0.0003;  % диаметр отверстия`
< 0.001	1000	4	`t = 0.00005;  % толщина отверстия`

< 0.001	1000	5	h = 0.0007; % высота отверстия
		6	
0.055	1000	7	format long
< 0.001	1000	8	height = 4.51e-3; % Размеры матрицы
< 0.001	1000	9	width = 2.88e-3;
< 0.001	1000	10	pxW = 752;
< 0.001	1000	11	pxH = 480;
< 0.001	1000	12	pxSize = width / pxW;
		13	
0.013	1000	14	angle = deg2rad(0 : 359); % вспомогательный массив углов
0.017	1000	15	defCircleX = (d/2) * cos(angle); % координата x контура пятна
0.009	1000	16	defCircleY = (d/2) * sin(angle); % координата y контура пятна
0.004	1000	17	xPx = ceil((defCircleX + width/2) / pxSize); % x - контур пятна в пикселях
0.004	1000	18	yPx = ceil((defCircleY + height/2) / pxSize); % y - контур пятна в пикселях
0.003	1000	19	defPxRadius = ceil((max(xPx) - min(xPx)) / 2);
		20	
< 0.001	1000	21	columns = 752;
< 0.001	1000	22	rows = 480;
0.003	1000	23	x = 1:columns;
0.002	1000	24	y = 1:rows;
		25	
< 0.001	1000	26	coloredX = [];
< 0.001	1000	27	coloredY = [];
		28	
< 0.001	1000	29	isFlag = false;
< 0.001	1000	30	isFlag2 = false;
		31	
< 0.001	1000	32	for i = 1 : columns
0.036	435379	33	for j = 1 : rows
520.624	208981920	34	if(squeeze(imageGS(j, i, :)) ~= 0)
0.370	1484025	35	coloredX(end + 1) = i;
0.255	1484025	36	coloredY(end + 1) = j;
0.053	1484025	37	isFlag2 = true;
0.052	1484025	38	if (length(coloredX) == 1)
< 0.001	893	39	isFlag = true;
0.052	1484025	40	end
6.759	208981920	41	end
6.964	208981920	42	end
0.034	435379	43	if (isFlag)
0.003	41762	44	if (~isFlag2)
< 0.001	892	45	break;
0.003	40870	46	end
0.029	434487	47	end
0.017	434487	48	isFlag2 = false;
0.033	434487	49	end
		50	
		51	% if (length(coloredX) ~= 0 && length(coloredY) ~= 0)
		52	
		53	% for i = 1 : length(coloredX)
		54	% imageProcessed(coloredY(i), coloredX(i), [2 2 2]) = 1;
		55	% end
		56	
< 0.001	1000	57	if (length(coloredX) == 0 || length(coloredY) == 0)
< 0.001	107	58	brutX = 0;
< 0.001	107	59	brutY = 0;
< 0.001	107	60	return;
< 0.001	893	61	end
0.002	893	62	brutX = ceil(sum(coloredX) / length(coloredX));
0.001	893	63	brutY = ceil(sum(coloredY) / length(coloredY));
		64	% if (ansX > 0 && ansX <= pxW)
		65	% if (ansY > 0 && ansY <= pxH)
		66	% imageProcessed(yAns1, xAns1, [2 1 2]) = 5;
		67	% end
		68	% end
		69	% end
0.010	893	70	end