hilbert_fractal_generating (Calls: 2, Time: 0.709 s)

Generated 20-May-2021 13:56:29 using performance time.
Function in file D:\Matlab\ITC_project\hilbert_fractal_generating.m
Copy to new window for comparing multiple runs

Parents (calling functions)

Function Name	Function Type	Calls
txt_stego_im	Function	1
im_stego_txt	Function	1

Lines that take the most time

Line Number	Code	Calls	Total Time (s)	% Time	Time Plot
13	`l(f-1,1)=isreal(z(f-1)-z(f));`	524286	0.160	22.6%	
46	`PX(n+1)=x;PY(n+1)=y;`	524286	0.132	18.6%	
15	`h(f-1,1)=1i*(z(f-1)-z(f));`	262144	0.079	11.2%	
22	`h(f-1,1)=-(z(f-1)-z(f));`	262142	0.068	9.6%	
14	`if(l(f-1,1)==0)`	524286	0.028	4.0%	
All other lines			0.241	34.0%	
Totals			0.709	100%	

Children (called functions)

No children

Code Analyzer results

Line Number	Message
13	The variable 'l' appears to change size on every loop iteration. Consider preallocating for…
15	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
17	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
19	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
22	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
24	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
26	The variable 'h' appears to change size on every loop iteration. Consider preallocating for…
35	If you intend to specify expression precedence, use parentheses () instead of brackets [].
35	If you intend to specify expression precedence, use parentheses () instead of brackets [].

Coverage results

Function listing

Time	Calls	Line	
		1	`function [PX,PY] = hilbert_fractal_generating(order)`
		2	`%creating hilbert fractal`
< 0.001	2	3	`a = 1 + 1i;`
< 0.001	2	4	`b = 1 - 1i;`
< 0.001	2	5	`z = 0;`
< 0.001	2	6	`for k = 1:order`
0.001	18	7	`    w = 1i*conj(z);`
0.008	18	8	`    z = [w-a; z-b; z+a; b-w]/2;`
0.001	18	9	`end`
		10	

< 0.001	2	12	for f=2:size(z,1)
0.160	524286	13	l(f-1,1)=isreal(z(f-1)-z(f));
0.028	524286	14	if(l(f-1,1)==0)
0.079	262144	15	h(f-1,1)=1i*(z(f-1)-z(f));
0.012	262144	16	if(h(f-1,1)>0)
0.005	131072	17	h(f-1,1)=1;
0.004	131072	18	else
0.005	131072	19	h(f-1,1)=0;
0.009	262144	20	end
0.009	262142	21	else
0.068	262142	22	h(f-1,1)=-(z(f-1)-z(f));
0.012	262142	23	if(h(f-1,1)>0)
0.005	131582	24	h(f-1,1)=1;
0.005	130560	25	else
0.005	130560	26	h(f-1,1)=0;
0.009	262142	27	end
0.018	524286	28	end
0.020	524286	29	end
		30	%plot(z);
0.001	2	31	h=logical(h);
		32	
< 0.001	2	33	x=2^order; y=1;
		34	
< 0.001	2	35	PX=[x];PY=[y];
< 0.001	2	36	for n=1:size(l,1)
0.026	524286	37	if(l(n,1) && h(n,1))
0.004	131582	38	y=y+1;
0.014	392704	39	elseif(l(n,1) && ~h(n,1))
0.005	130560	40	y=y-1;
0.013	262144	41	elseif(~l(n,1) && h(n,1))
0.004	131072	42	x=x-1;
0.004	131072	43	else
0.004	131072	44	x=x+1;
0.017	524286	45	end
0.132	524286	46	PX(n+1)=x;PY(n+1)=y;
		47	
0.019	524286	48	end
		49	
		50	%save("hilbert_order"+string(order)+".mat",'PX','PY')
< 0.001	2	51	end