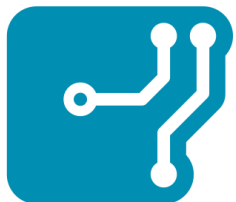


U.PORTO

FC FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

U.PORTO *Jr*



INESCPORTO

INSTITUTO DE ENGENHARIA DE SISTEMAS
E COMPUTADORES DO PORTO
LABORATÓRIO ASSOCIADO

2009 AGO 31 – SET 5

Escola de Verão de Física

Sistema de Varrimento Laser para Caracterização Tridimensional de Objectos

Alexandre Morais

Ana Peixoto

Casimiro Ferreira

Paula Ferradaz

Pedro Rodrigues

Objectivos

LASER

Imagens digitais

Física

Resolução

Matemática

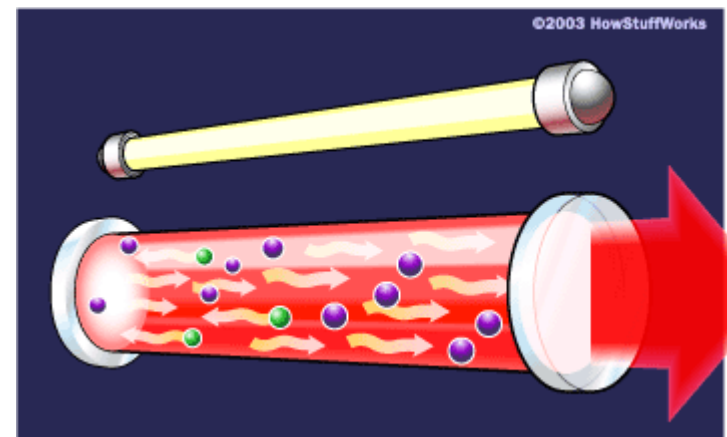
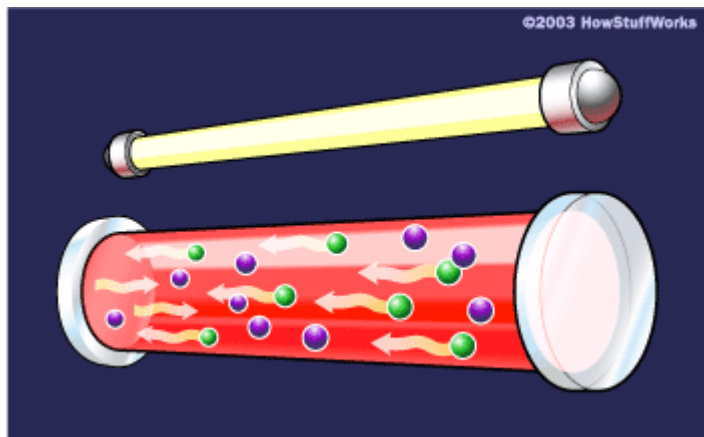
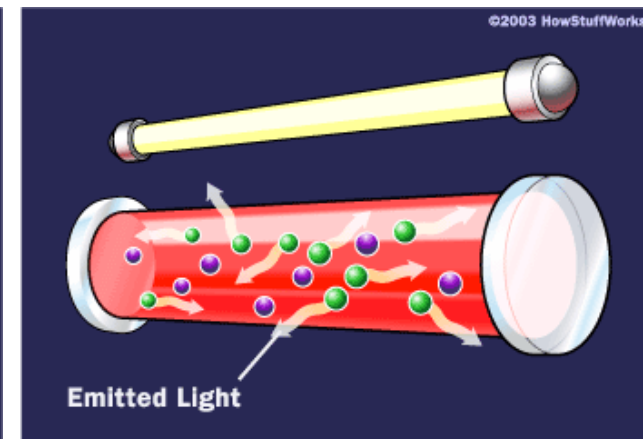
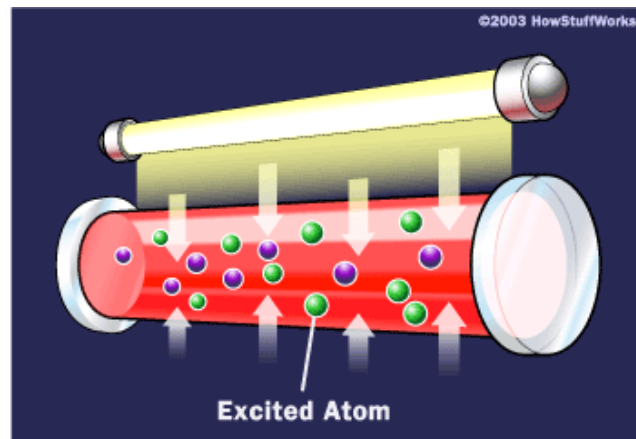
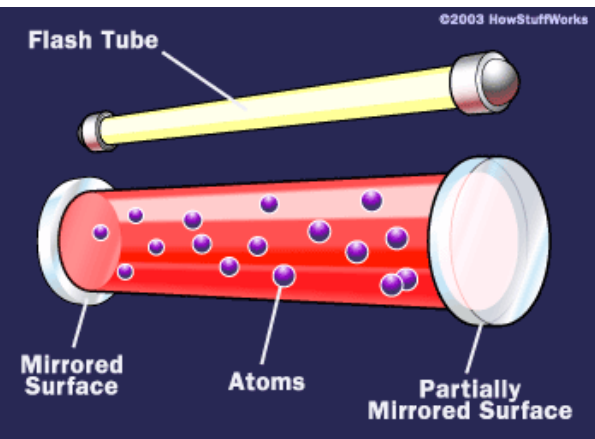
Profundidade de cor

Programação

Processamento de imagem

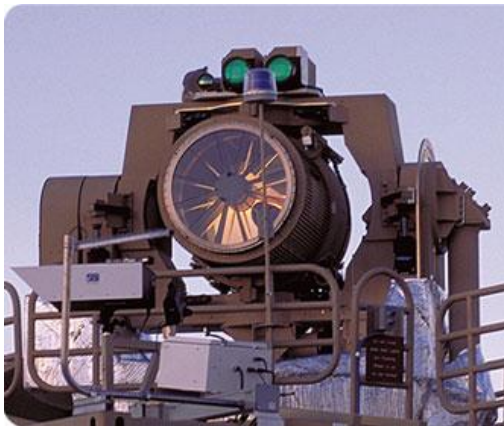
Scanner 3D

LASER - Light Amplification by Stimulated Emission of Radiation

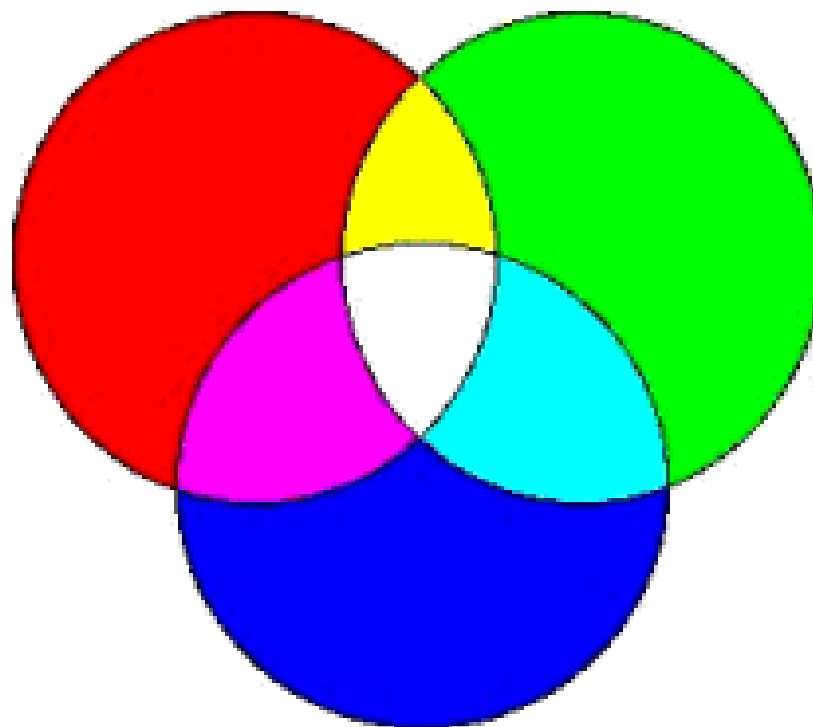


LASER

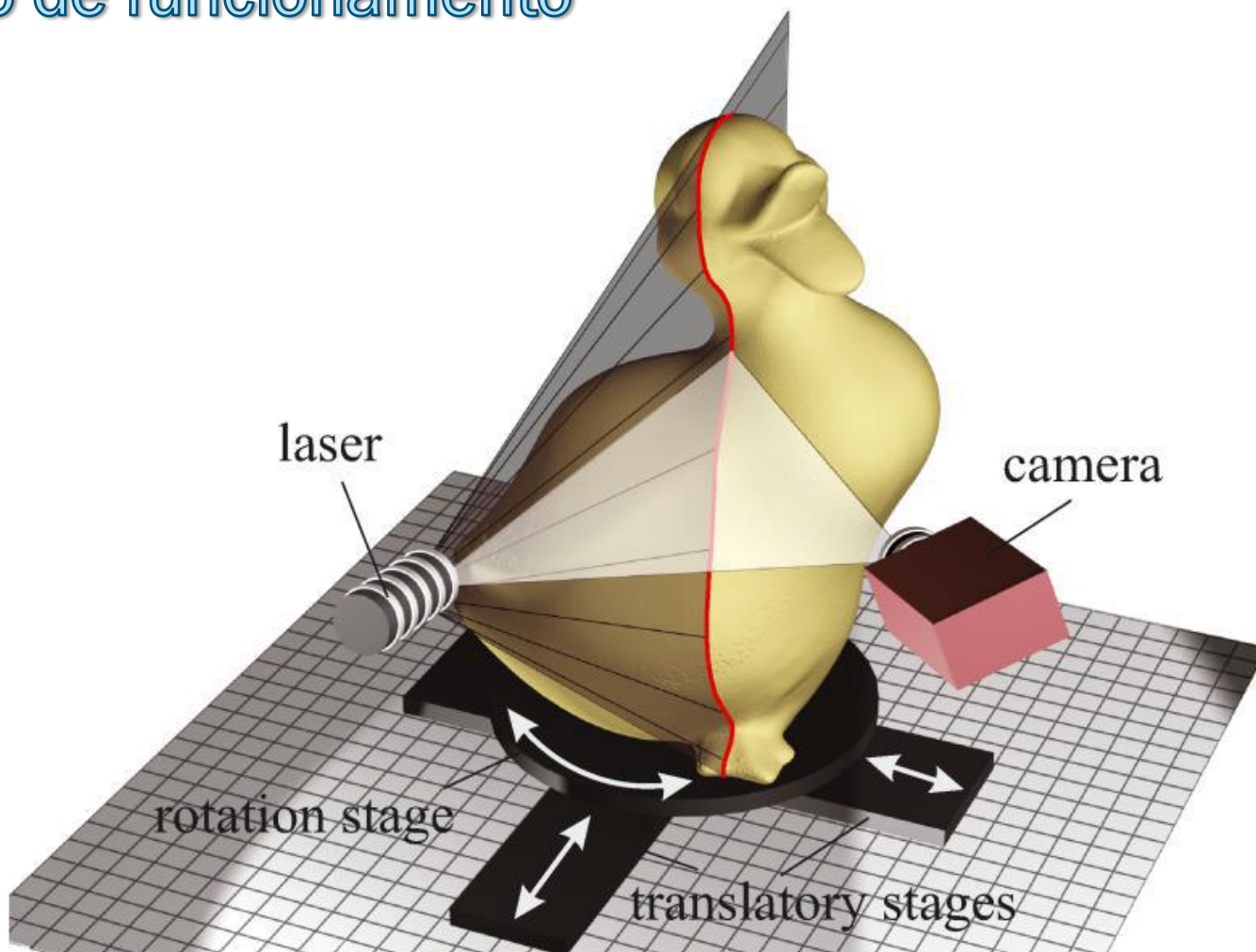
Aplicações ?????



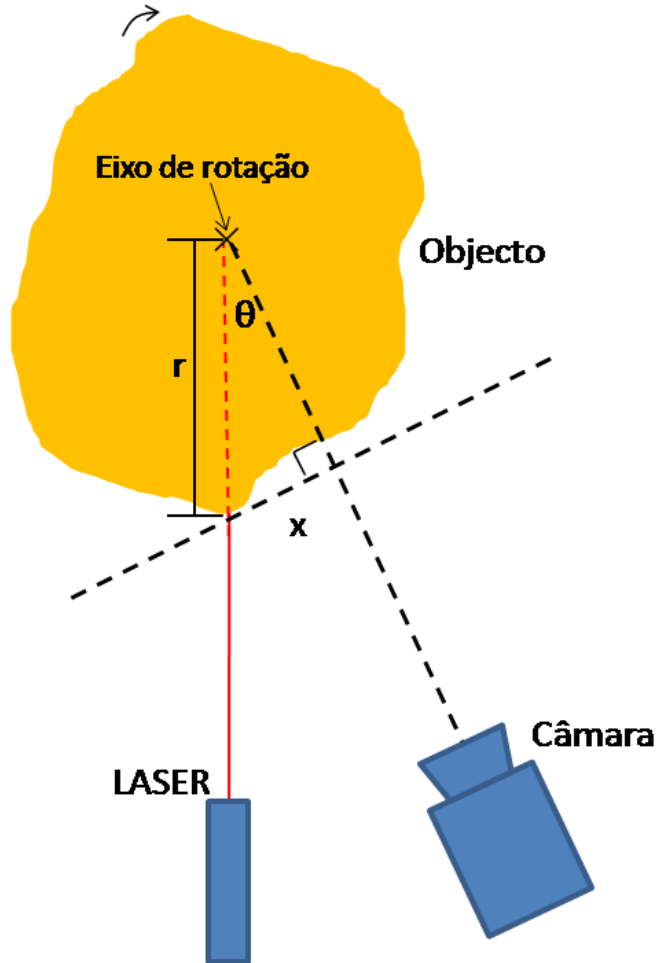
Formação de imagens



Princípio de funcionamento

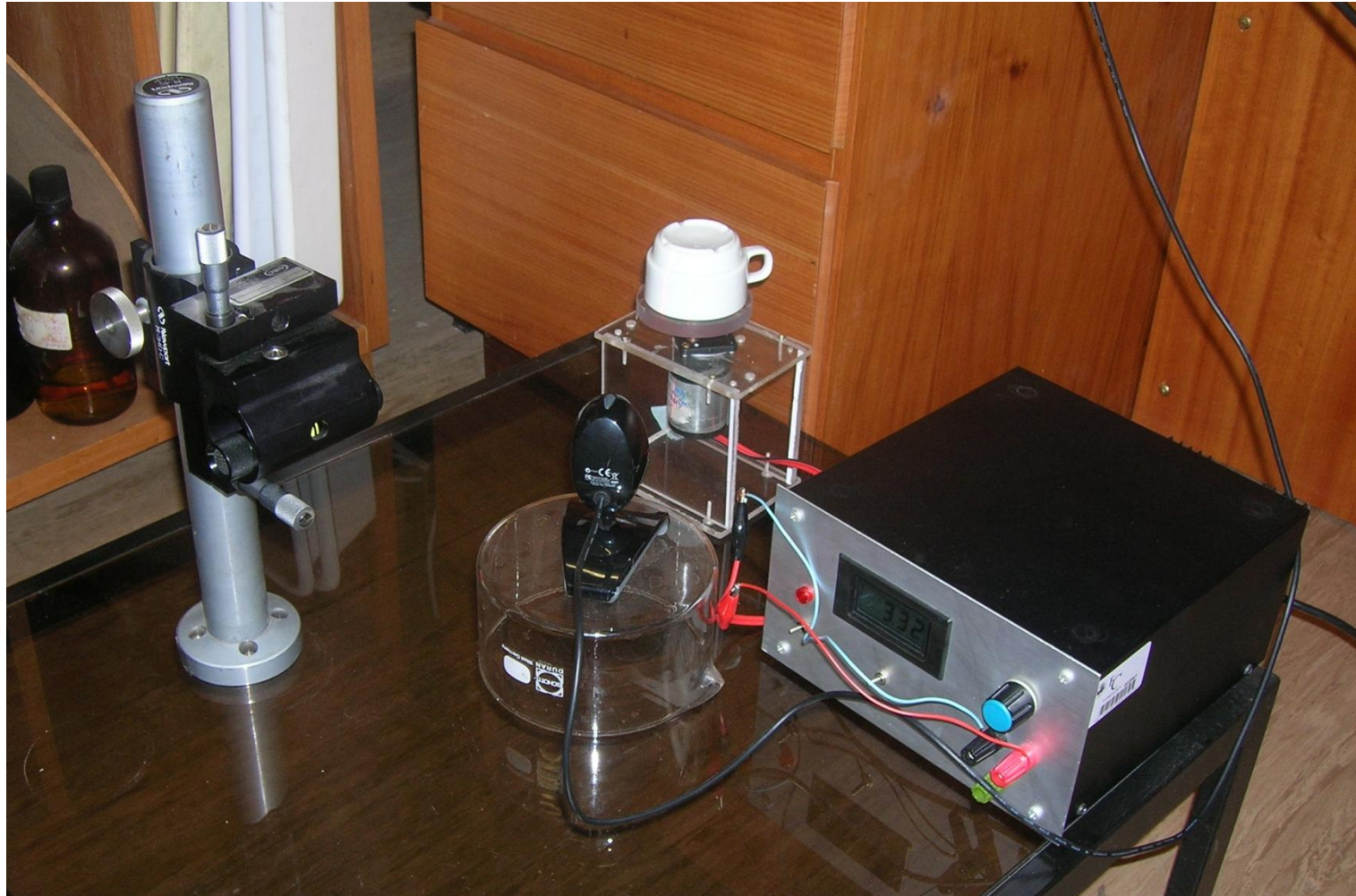


Princípio de funcionamento

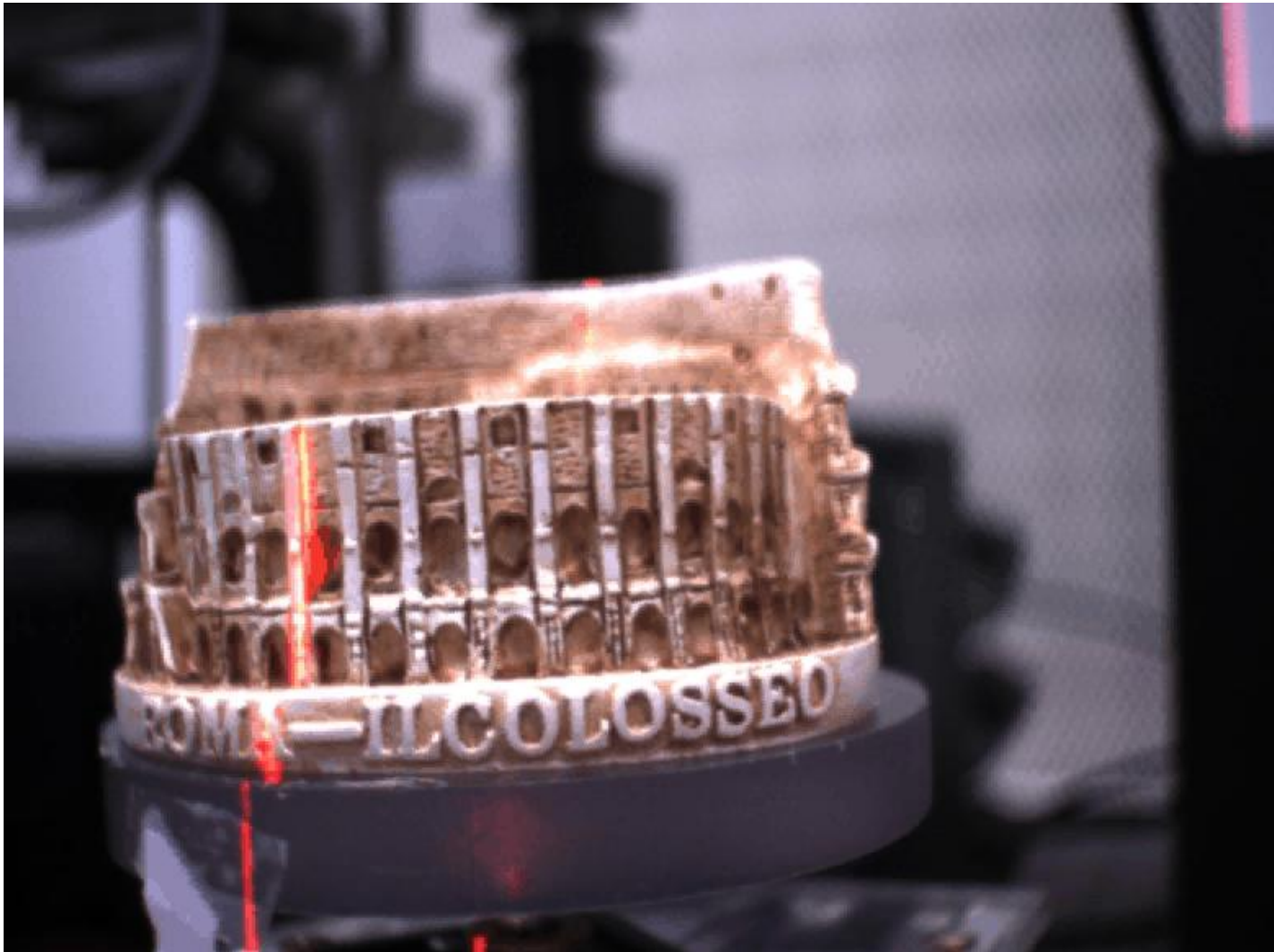


$$r \sin \theta = x \Leftrightarrow r = \frac{x}{\sin \theta}$$

Montagem Experimental



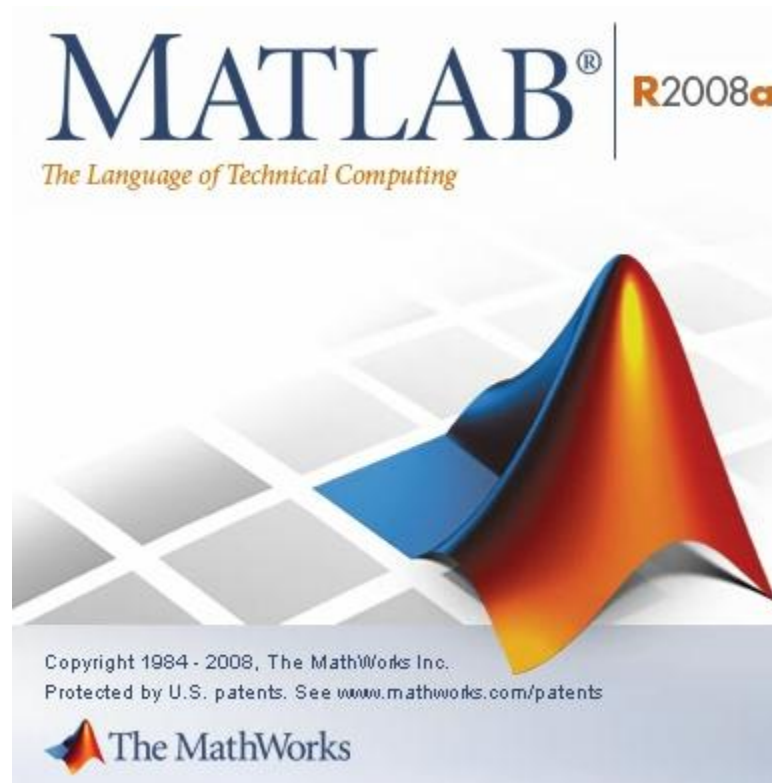
Introdução



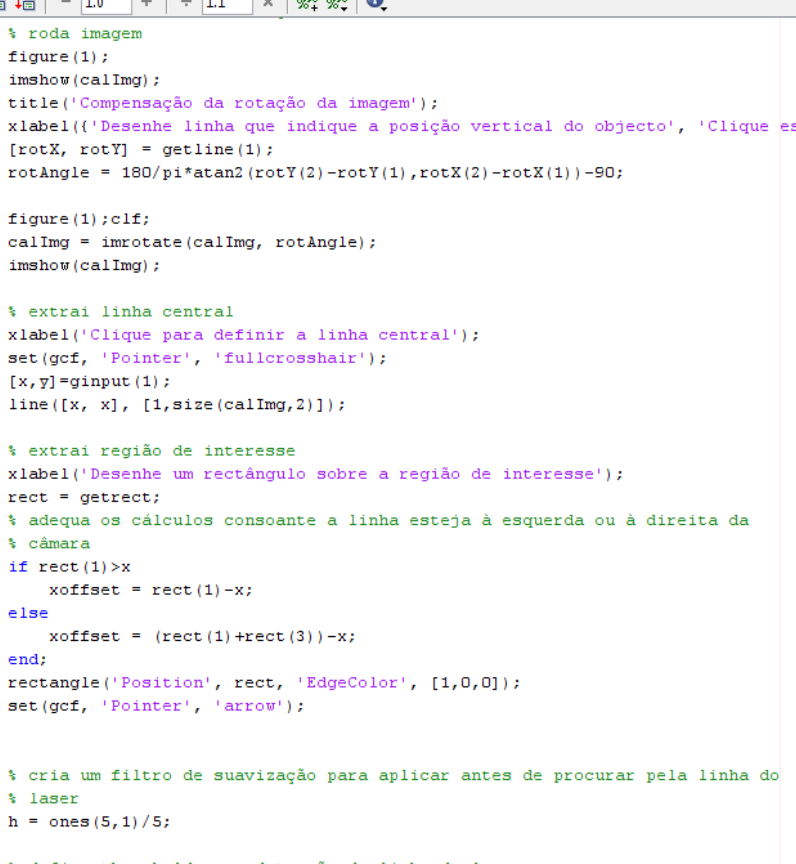
Programação de computadores

Desenvolvimento de algoritmos (sequências de instruções lógicas) que nos permitem resolver um determinado problema.

Alguns exemplos de linguagens de programação: C/C++, Pascal, Fortran, Python, Perl, HTML, Assembly, PHP, LabVIEW, Java, Prolog, etc.

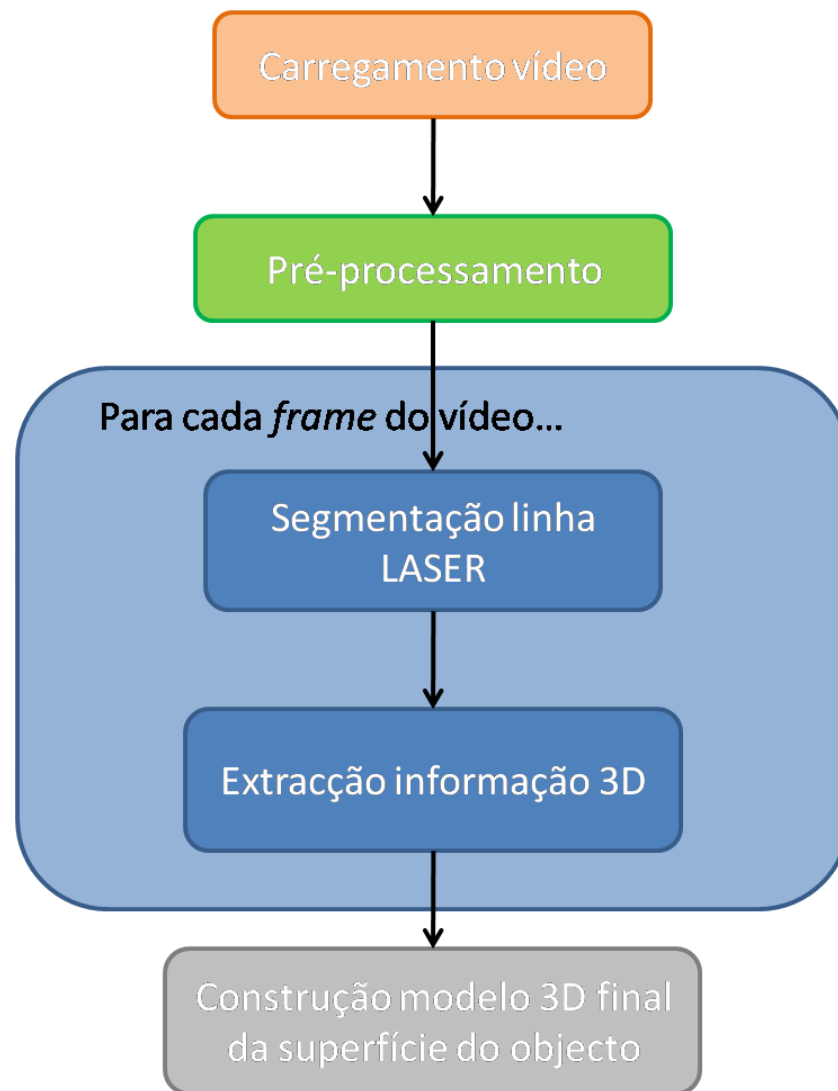


Programação de computadores



```
Editor - D:\FEUP\PDVEEC\2 Semestre\RAO3D\Varrimento_laser.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Stack: Base [Icons]
- 1.0 + ÷ 1.1 x % % % % %
44 % roda imagem
45 figure(1);
46 imshow(calImg);
47 title('Compensação da rotação da imagem');
48 xlabel('Desenhe linha que indique a posição vertical do objecto', 'Clique esquerdo');
49 [rotX, rotY] = getline(1);
50 rotAngle = 180/pi*atan2(rotY(2)-rotY(1), rotX(2)-rotX(1))-90;
51
52 figure(1);clf;
53 calImg = imrotate(calImg, rotAngle);
54 imshow(calImg);
55
56 % extrai linha central
57 xlabel('Clique para definir a linha central');
58 set(gcf, 'Pointer', 'fullcrosshair');
59 [x,y]=ginput(1);
60 line([x, x], [1,size(calImg,2)]);
61
62 % extrai região de interesse
63 xlabel('Desenhe um rectângulo sobre a região de interesse');
64 rect = getrect;
65 % adequa os cálculos consoante a linha esteja à esquerda ou à direita da
66 % câmara
67 if rect(1)>x
68     xoffset = rect(1)-x;
69 else
70     xoffset = (rect(1)+rect(3))-x;
71 end;
72 rectangle('Position', rect, 'EdgeColor', [1,0,0]);
73 set(gcf, 'Pointer', 'arrow');
74
75
76 % cria um filtro de suavização para aplicar antes de procurar pela linha do
77 % laser
78 h = ones(5,1)/5;
79
80 % define threshold para detecção da linha do laser
81 threshold = 0.1;
82
```

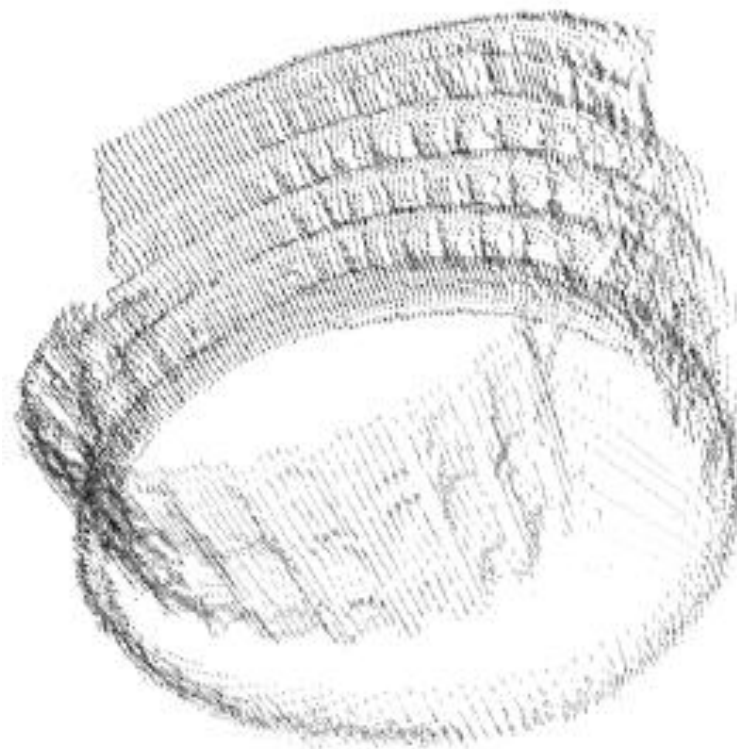
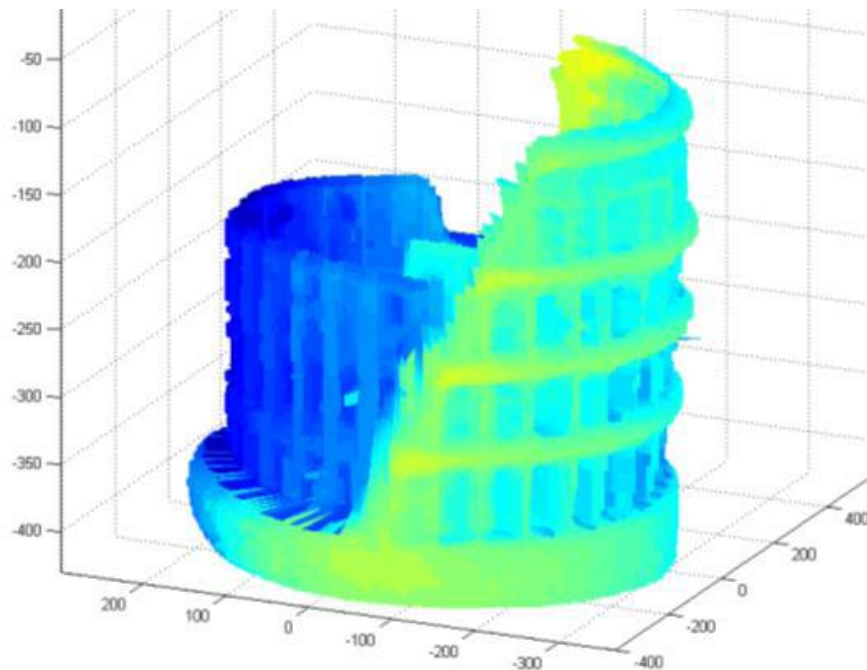
Software



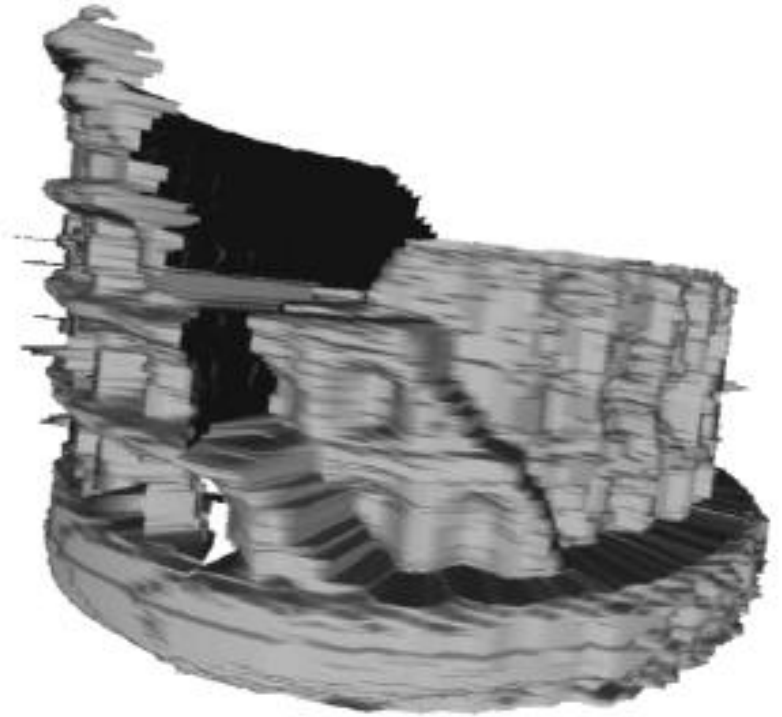
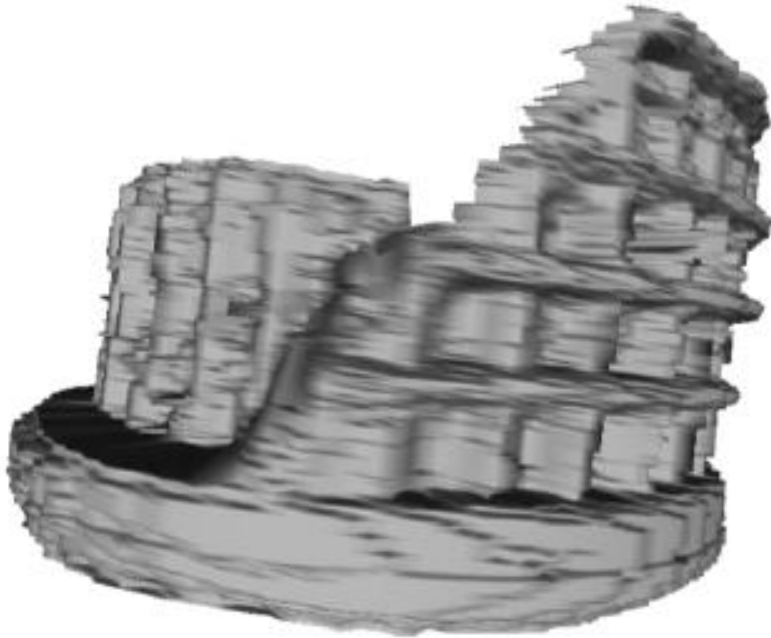
Resultados



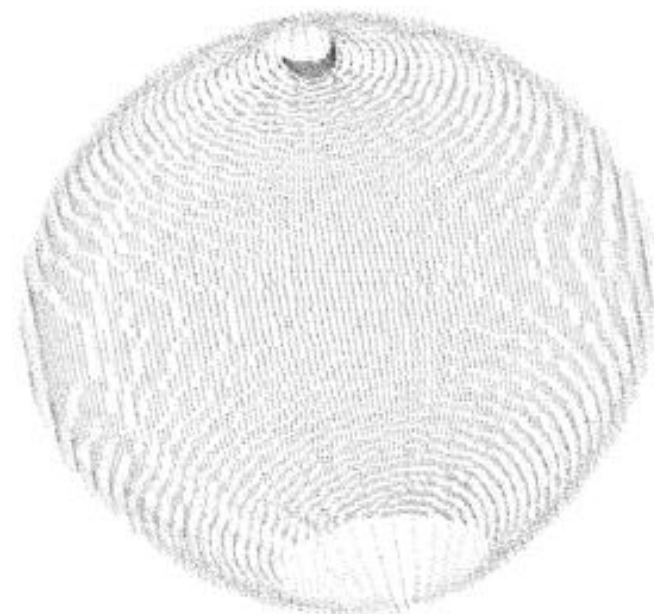
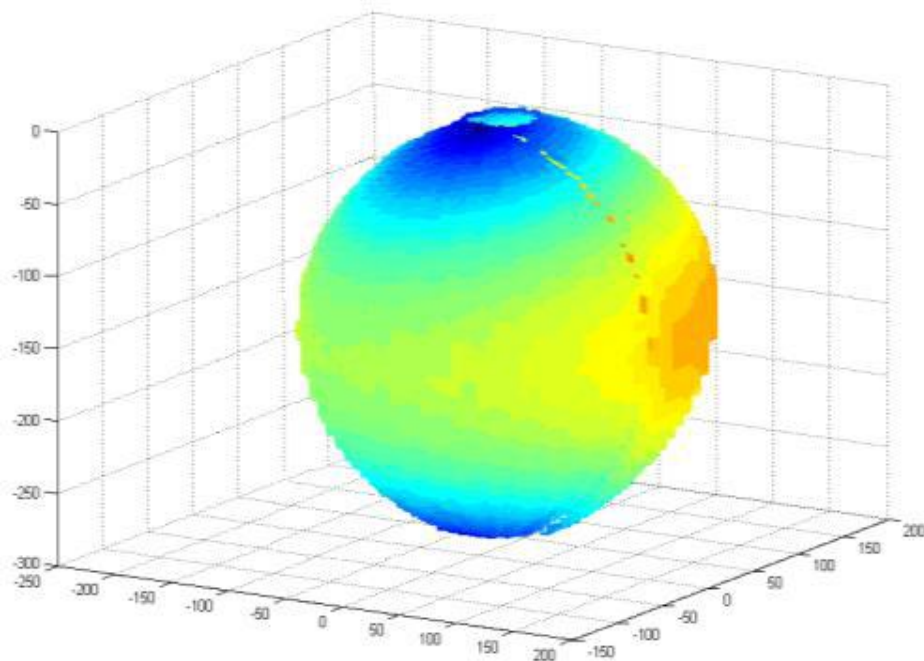
Resultados



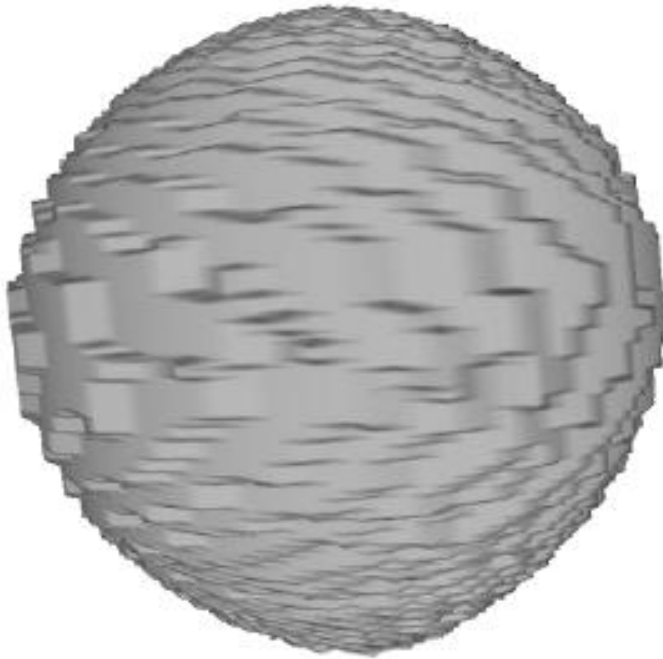
Resultados



Resultados



Resultados



Análise e discussão de resultados

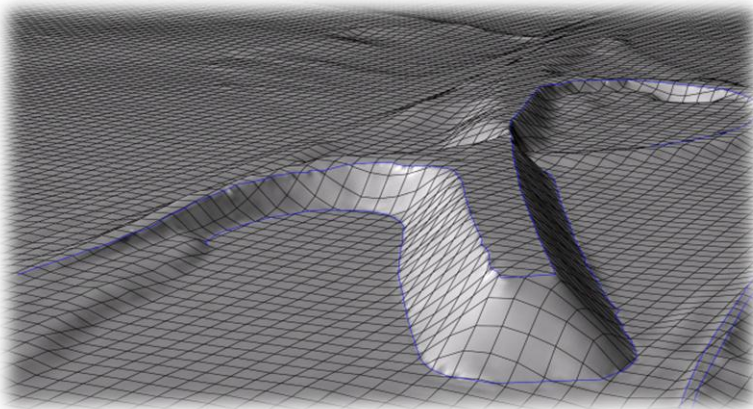


- Simplicidade
- Custo reduzido
- Visualização 3D de objectos em 360°
- Qualidade dos resultados



- Não tem a capacidade de operação em tempo-real
- Objectos translúcidos, transparentes ou espelhados limitam severamente a qualidade dos resultados
- Oclusões devido à forma do objecto limitam o desempenho deste método
- Principal limitação vem da espessura do feixe laser na imagem

Aplicações



Agradecimentos

- INESC Porto
- Departamento de Física – FCUP
- Universidade Júnior
- CFP
- Filipe Magalhães

Perguntas? Questions? Preguntas? Fragen? Otázky? Kérdésre?

