



Disciplina: Processamento de Imagens

Turma: T02 2021.2

Professora: Beatriz Trinchão Andrade

1. Especificação¹

Você e seu grupo foram contratados pela empresa GenericStuff para trabalhar em um sistema de inspeção de objetos. Este sistema captura imagens contendo objetos de duas categorias: objetos com furos, e objetos sem furos. O seu trabalho é implementar um algoritmo que informe, para cada imagem capturada pelo sistema, o total de objetos na imagem, e quantos objetos com furos e sem furos a imagem contém.

Assumam que os objetos passam sobre uma esteira rolante e uma câmera captura imagens binárias, como a ilustrada abaixo. Os objetos não se tocam nem estão sobrepostos, mas podem estar próximos uns dos outros. Os objetos podem ter qualquer forma ou tamanho. Caso o grupo julgue necessário, poderá fazer asserções sobre o tamanho ou forma dos objetos e dos furos.

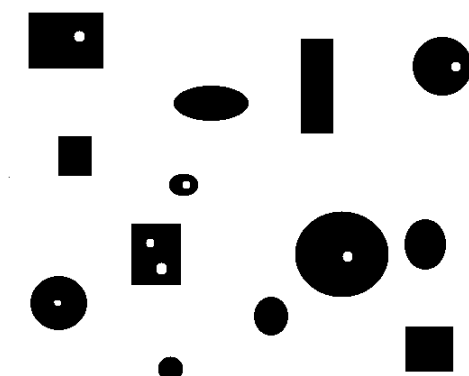


Figura 1: Exemplo de imagem de entrada.

A entrada do seu algoritmo será uma imagem capturada pelo sistema descrito acima, representada como uma imagem binária no formato PBM ASCII (PGM tipo P1). Nesta imagem, os pixels que correspondem ao fundo (aparecem como brancos na Figura 1) terão valor **0** e os pixels que correspondem aos objetos (pixels pretos da Figura 1) terão valor **1**. Essa é uma forma de encarar a imagem como uma máscara, que informa 1 caso um pixel pertença ao objeto de interesse (o que seria equivalente ao booleano true) ou 0 caso contrário (aqui, seria equivalente ao false). Mais detalhes sobre essa imagem e o que espero como seu conteúdo foram dados na aula do dia 11/04/2022. A gravação desta aula está na turma do Classroom.

Como saída, seu programa retornará o total de objetos na imagem, e quantos objetos com furos e sem furos a imagem contém. O sistema deve funcionar no modo console, recebendo a imagem como entrada na linha de comando e imprimindo os dados obtidos no console. Cada grupo deve enviar no mínimo 3 imagens PBM de teste.

¹ Este trabalho foi inspirado em uma questão de exercício da disciplina “Designing Applications that See” (CS377S), da Universidade de Stanford. 2008.

A GenericStuff é uma empresa um tanto peculiar, e implementa uma metodologia “de baixo para cima” no desenvolvimento dos seus produtos. Desta forma, vocês também precisam, além de implementar o algoritmo descrito acima, dar uma utilidade para ele. Ou seja, vocês precisam imaginar um potencial de uso para o método que vocês desenvolveram e, aproveitando que vocês já estão embalados nesse tema, analisar como a aplicação proposta se enquadra em sua noção de **ética**.

2. Relatório

O relatório deve ser composto por duas partes, descritas a seguir.

Primeira parte:

- Com base no que foi solicitado, o que foi implementado pelo grupo;
- Quais as técnicas aprendidas na disciplina foram aplicadas e que parâmetros foram usados;
- As soluções que vocês desenvolveram para os principais problemas encontrados;
- Se foram necessárias asserções sobre o tamanho ou forma dos objetos. Caso sim, explicitar quais foram.
- Instruções sobre como compilar e executar o projeto no Linux.

Segunda parte:

- Que potencial de uso vocês imaginam para o algoritmo desenvolvido;
- Que adaptações seriam necessárias para que sua ideia funcione;
- Como sua ideia se encaixa no conceito de ética estudado pelo grupo.

O uso de imagens para ilustrar conceitos é altamente recomendado.

3. Avaliação

A avaliação será feita a partir:

- Relatório
- Código fonte
- Imagens de teste enviadas pelo grupo (envio mínimo de 3 imagens)
- Possíveis entrevistas ou apresentações dos grupos

Critérios Avaliados

- 3.5: atendimento às especificações do trabalho
- 1.0: aplicação dos conceitos vistos em aula
- 1.0: eficiência da solução encontrada
- 1.0: instruções de compilação e execução
- 0.5: organização e documentação do código fonte
- 1.0: primeira parte do relatório
- 2.0: segunda parte do relatório
- até 1.0 extra (robustez e/ou criatividade)

4. Importante

- O trabalho prático deverá ser feito em grupos de até três pessoas. Por padrão, os grupos serão os mesmos do Trabalho de Pesquisa. Caso queiram mudar a composição dos grupos, escrever para beatriz.trinchao@academico.ufs.br até o dia 21/04/2022 informando os nomes.
- Qualquer suspeita de plágio resultará em nota zero para todos os envolvidos.
- Prazos de entrega do código-fonte, imagens de teste e relatório: vide tarefa no SIGAA. O projeto, as imagens e o relatório devem ser enviados em um arquivo comprimido pelo SIGAA.
- Podem ser agendadas apresentações para cada grupo.
- Linguagens aceitas: Python, Java, C, C++. Não usar bibliotecas de processamento de imagens ou visão computacional prontas.
- Entregas após o prazo sofrem penalidade de metade da nota da tarefa por dia de atraso. Trabalhos entregues com atraso devem ser submetidos na Tarefa 'Entrega após prazo', no SIGAA, que ficará aberta durante todo o período. Arquivos enviados por e-mail não serão considerados. Caso as imagens não caibam no envio, podem mandar o link da pasta onde elas estão no relatório.
- A depender do andamento da disciplina podem haver alterações no prazo.

Bom trabalho!