
Relatório Final - Projeto da disciplina de Processamento de Imagens (SCC0251)

Kauê Lopes De Moraes - 9277576
Lucas Kassouf Crocomo - 8937420

Esteganografia no domínio das frequências

28 de Junho de 2018

Proposta inicial

A ideia inicial do projeto era usar técnicas aprendidas ao longo do semestre para fazer um trabalho na área de esteganografia. Basicamente a ideia é passar a imagem do Domínio da Imagem para o Domínio das Frequências, e assim trabalhar nas frequências mais altas para poder esconder uma mensagem. A ideia surgiu ao lembrarmos do funcionamento da compressão feita no formato JPEG, que usa a transformada de Cossenos e elimina de forma inteligente os últimos coeficientes, que usualmente são mais próximos ao zero. No caso, ao invés de apagar esses últimos, trabalharíamos para codificar uma mensagem.

Modificando as últimas frequências, há uma alteração da imagem, ficando ruidosa. Será investigado alguns filtros, como filtro adaptativo da média, mediana e filtro convencional Gaussiano. Além de utilizar esses filtros, será aferido o desempenho de cada um para efeito de comparação. A comparação será feita através do cálculo do RMSE da imagem original, filtrada e após processamento (esteganografia).

Desenvolvimento e erros encontrados

Primeiramente, concluímos que não seria adequado a utilização de RMSE para mensuração de similaridade entre as imagens originais e processadas (demonstração no notebook jupyter).

Como já abordado, a proposta inicial era esconder uma mensagem no domínio da frequência de uma imagem utilizando as frequências mais altas (menos relevantes para a visualização da imagem). Para isso, realizamos o FFT nas imagens as quais estão no domínio real (0-255) e, dessa forma obtemos a imagem no domínio das frequências (representadas por números complexos). Testamos vários filtros para tentar suavizar as frequências mais altas, como por exemplo: bilateral, wavelet, total variance *chambollet* e gaussiano. Após isso, aplicou-se uma máscara com intuito de zerar o coeficientes das frequências mais altas. Então, foi realizada a inserção da mensagem nesses coeficientes vazios. Assim, aplicamos IFFT para obtermos a imagem com a mensagem esteganografada. Nessa parte, é importante ressaltar que é necessário a igualdade de todos os coeficientes geradas pela função de IFFT para que possamos retornar à nossa imagem original. Dada essa informação e o fato de que para salvar a imagem num arquivo é necessário utilizarmos np.uint8, notamos uma **impossibilidade** de recuperar a mensagem, por causa dos erros de conversão de tipos de dados complexos e inteiros.

Todos esses processos são encontrados em notebooks Jupyter. O pré-processamento das imagens e geração dos filtros pode ser encontrado [aqui](#). Os passos seguintes, como maiores manipulações da imagem citadas acima, são encontradas [aqui](#).

Por conta de não termos conseguido concluir totalmente o projeto, pelo empecilho encontrado na hora de recuperarmos a mensagem esteganografada, não se encontrou uma maneira de disponibilizar um arquivo .py para demonstração. Entretanto, como já mencionado, todo o **passo-a-passo** (e **comentários**, inclusive) utilizado para desenvolvimento do trabalho pode ser encontrado [nesse repositório](#).