

# Project X Writeup

## Instructions

- Describe any interesting decisions you made to write your algorithm.
- Show and discuss the results of your algorithm.
- Feel free to include code snippets, images, and equations.
- Use as many pages as you need, but err on the short side If you feel you only need to write a short amount to meet the brief, th
- **Please make this document anonymous.**

## In the beginning...

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. See Equation 1.

$$a = b + c \tag{1}$$

## Interesting Implementation Detail

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

My code snippet highlights an interesting point.

```
1 one = 1;  
2 two = one + one;  
3 if two == 2  
4     disp( 'This computer is not broken.' );  
5 end
```

## A Result

1. Result 1 was a total failure, because...
2. Result 2 (Figure 1, left) was surprising, because...
3. Result 3 (Figure 1, right) blew my socks off, because...

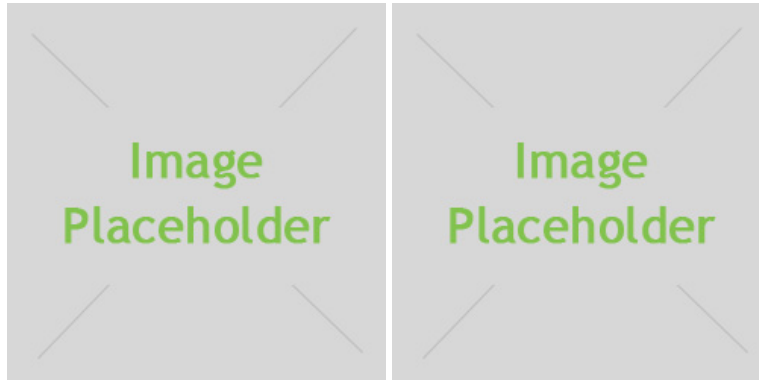


Figure 1: *Left:* My result was spectacular. *Right:* Curious.

My results are summarized in Table 1.

Condition	Time (seconds)
Test 1	1
Test 2	1000

Table 1: Stunning revelation about the efficiency of my code.

## Considerações exercício 1

Para o primeiro exercício foi utilizado o filtro e a função (*my\_imfilter*) para calcular a imagem1 em baixa frequência.

Para calcularmos a imagem2 em alta frequência, foi utilizado o mesmo filtro utilizado na imagem1, após a utilização do filtro obtemos uma nova imagem2filtrada em baixa frequência, usamos a diferença entre a imagem2 original com a imagem2filtrada, para obtermos a imagem2 sem suas componentes de baixa frequência, desta forma, obtendo a imagem2 em alta frequência.

Por fim é realizada a junção das duas imagens para a obtenção da imagem híbrida.

## Considerações exercício 2

No exercício número 2, a imagem1 já estava calculada com suas componentes em baixa frequência.

Primeiramente o atributo *cutoff\_frequency* foi modificado para a obtenção dos melhores resultado, esse atributo foi sendo modificado de 10 em 10 e por fim determinados o valor dele em 20.

Para calcular a imagem2 em alta frequência, foi utilizado o mesmo código utilizado na imagem1, com apenas uma modificação, o filtro utilizado na imagem1 foi invertido na tentativa de obtermos a imagem2 em alta frequência, para isto, foi utilizado a função (*imcomplement*) do octave e após a utilização do filtro invertido foi detectado que a imagem2 estava com suas componentes em alta frequência.

Por fim, a junção das duas imagens foi realizada para a obtenção da imagem híbrida.