

Relatório trabalho Filtragem de Imagens

Os métodos aplicados foram média[0], mediana[1] e empilhamento[2] em cima de imagens com ruído aleatório *salt and pepper*.

```
niveis De Ruído = [0.01, 0.02, 0.05, 0.07, 0.1]
valor filtro = [0, 1, 2]
```

```
python3 filtro.py imagemEntrada nivelRuido valorFiltro
imagemSaida
```

PSNR médio dos ruídos	
Nível de ruído	Média PSNR
0.01	21,95
0.02	18,97
0.05	14,97
0.07	13,50
0.1	11,97

tabela 1: média PSNR ao aplicar diferentes níveis de ruído

Durantes as experimentações o menor nível de ruído(0.01) mostrou-se o com o maior PSNR após os filtros e só com o ruído, entretanto, dependendo do método não importou o nível de ruído pois os valores obtidos foram muito próximos, sendo assim para este tipo de ruído o método de empilhamento foi notório embora o custo de tempo. no gráfico abaixo o método fica visivelmente acima nos menores níveis de ruído, porém ao final perde para o método de mediana.

PSNR médio com filtros			
Nível de ruído	média	mediana	empilhamento
0.01	24,05	27,24	36,37
0.02	24,18	27,13	32,74
0.05	22,37	26,71	27,81
0.07	21,54	26,25	25,71
0.1	20,23	25,18	23,32

tabela 2: PSNR médio entre os métodos para cada nível de ruído

média, mediana e empilhamento

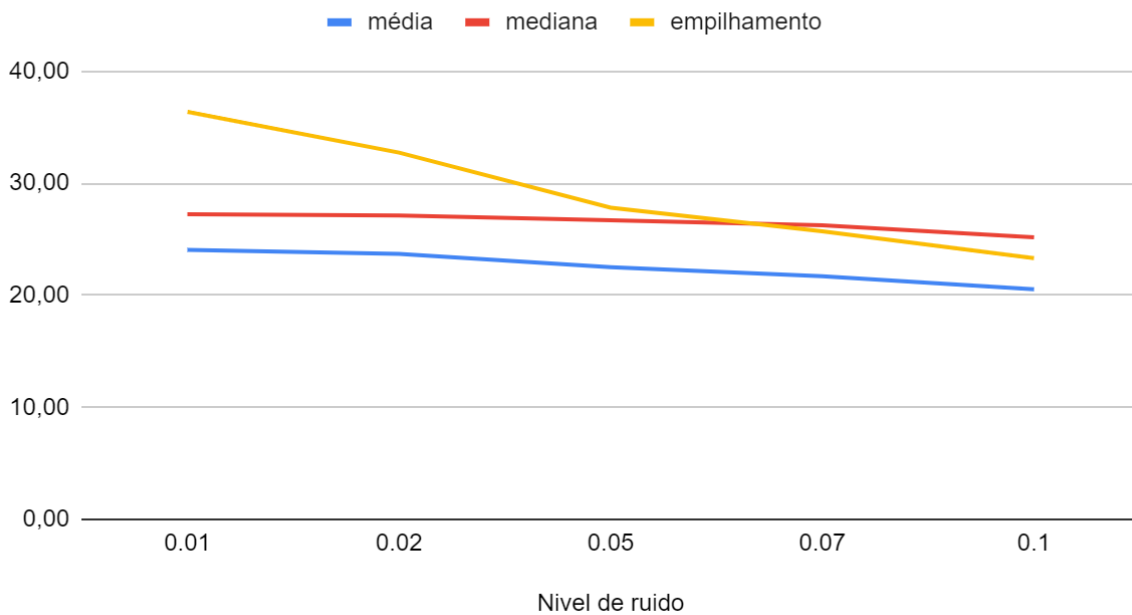


grafico 1: grafico da tabela 2

melhor método por nível de ruído	
Nível de ruído	Melhor método
0.01	empilhamento
0.02	empilhamento
0.05	empilhamento
0.07	mediana
0.1	mediana

tabela 3: melhor método(PSNR mais alto) por nível de ruído

Em seguida será demonstrado dados particulares de cada metodos:

Dados da utilização do método de mediana:

medianBlur(imagemRuido, x) PSNR para diferentes valor de x						
Nivel de ruido	X=1	X=3	X=5	X=7	X=13	X=27
0.01	21,99	27,24	24,59	23,70	22,49	23,70
0.02	18,95	27,14	24,56	23,69	22,49	23,69
0.05	14,94	26,73	24,48	23,65	22,48	23,65
0.07	13,52	26,33	24,41	23,61	22,47	23,61
0.1	11,99	25,09	24,29	23,57	22,44	23,57

tabela 4: comparação de diferentes valores de x sendo x ímpar parâmetro para a função medianBlur da biblioteca opencv e seus devidos PSNRs

Dados da utilização do método de média:

blur(imagemRuido, (x, y)) PSNR para diferentes valor de x e y							
Nivel de ruido	(2, 2)	(10, 10)	(20, 20)	(1, 3)	(5, 3)	(7, 3)	(7, 5)
0.01	24,49	22,66	21,56	24,57	24,84	24,30	23,75
0.02	22,87	22,54	21,48	22,43	24,21	23,87	23,47
0.05	19,86	21,96	21,10	19,03	22,48	22,51	22,49
0.07	18,54	21,46	20,74	17,67	21,52	21,62	21,79
0.1	16,99	20,63	20,09	16,13	20,14	20,41	20,68

tabela 5: comparação de diferentes valores de x e y sendo x e y parâmetro dado como para a função blur da biblioteca opencv e seus devidos PSNRs

Dados da utilização do método de empilhamento:

PSNR para diferentes numero de vezes repetindo à diferente quantidades de sobreposições								
Nível de ruído	4*13	5*20	4*20	5*50	5*10	2*100	2*20	4*100
0.01	36,38	37,50	33,67	37,09	27,60	15,99	16,25	28,41
0.02	32,95	35,02	31,34	34,22	27,51	15,78	16,02	27,51
0.05	27,81	29,75	27,11	29,06	26,45	15,19	15,38	25,14
0.07	25,70	27,37	25,20	26,81	25,41	14,82	14,98	23,80
0.1	23,35	24,67	23,02	24,27	23,69	14,30	14,42	22,08

tabela 6: comparação de diferentes valores para a repetição de diferentes valores de sobreposições e seus devidos PSNRs por nível de ruído

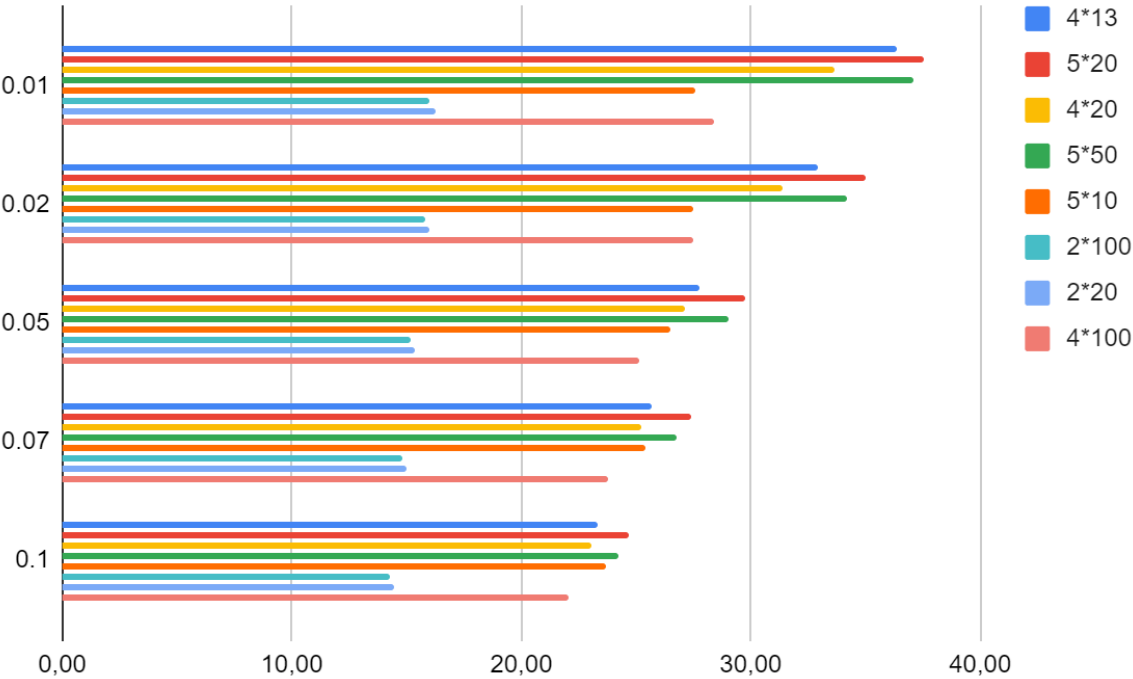


gráfico 2: gráfico gerado através da tabela 6