



Compte rendu 5

BELDJILALI Maxime, CHATEAUNEUF Arthur

Mise en pratique des filtres	2
Utilisation	2
Bruitage par carte de fréquence	3
Bruitage par fréquence	3
Pistes infructueuses	4
Conclusion	4
Annexes	5
Références	5
Dépot Github	5

Mise en pratique des filtres

Utilisation sur les méthodes naïves

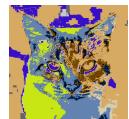
Nous avons mis en place le chargement de l'intégralité des images d'un fichier pour analyse du CNN dessus. Cela nous permet de pouvoir rapidement comparer l'impact de chacun de nos ajouts aux filtres lorsque nous en composons.

Voici les résultats que nous obtenons sur les filtres naïfs :



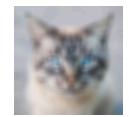
Précision : 0.46

NAIVE LSB CUT



Précision : 0.23

NAIVE LSB CUT [YCrCb]



Précision : 0.9

NAIVE HEAVY BLUR



Précision : 0.11

NAIVE WAVE DIVIDE



Précision : 0.19

NAIVE LINE NOISE

Nous remarquons que le floutage améliore grandement la précision du modèle. Nous pensons que cela provient du fait que ce filtre effectue une meilleure extraction des basses fréquences que le prétraitement que notre CNN possède actuellement.

De plus, couper les basses fréquences, comme visible par le filtre NAIVE LSB CUT ne donne que des résultats faibles (comparé à la perte de qualité de l'image) mais pouvant être améliorer en dégradant énormément les couleurs (même filtre en YCrCb).

Les meilleurs résultats sont visibles lorsque nous attaquons les grandes et moyennes fréquences. La précision de NAIVE WAVE DIVIDE, quant à elle, est si proche du résultat aléatoire (10% pour 10 classes) que nous la considérons nulle pour le CNN, c'est à dire que le CNN devine la classe avec la même efficacité que l'aléatoire.

Bruitage par carte de fréquence

Bruitage par fréquence

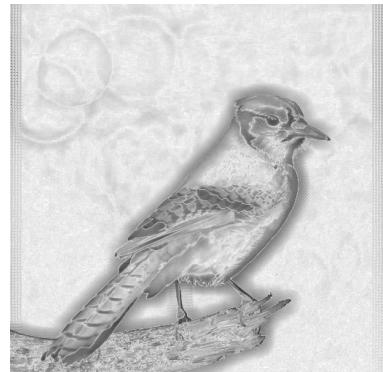
Nous avons vu que le bruitage et l'attaque des basses fréquences sont très efficaces, nous souhaitons combiner ces deux aspects afin d'effectuer un bruitage sur une image en fonction de la contribution de fréquence de chaque pixel. Pour cela, nous prenons l'image originale ainsi qu'une version floutée et comparons les différences pixels à pixels entre le flou et l'originale :



Image originale

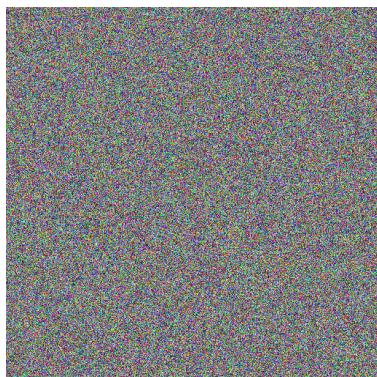


Image floutée



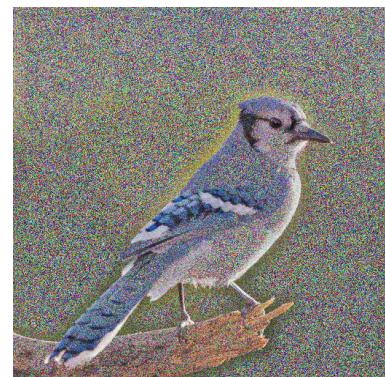
Carte des fréquences

Nous obtenons ainsi une carte de fréquence que nous utilisons en entrée d'une fonction de transparence entre l'image originale et un bruit aléatoire. Cela nous permet de bruiter la majorité de l'information de l'image, tout en gardant le sujet visible pour un observateur humain :



Bruitage sans carte de fréq

Précision CNN : ~10% (nulle)



Bruitage avec carte de freq

Précision CNN : ~10% (nulle)

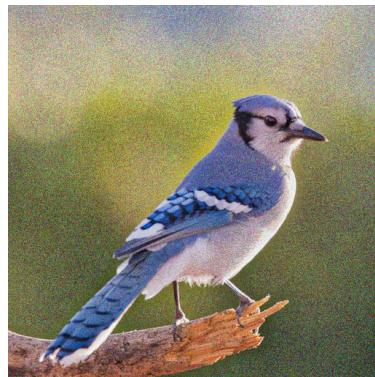
Optimisation de l'impacte visuel

Nous avons ainsi obtenu un filtre supprimant totalement la reconnaissance de notre CNN tout en garantissant une reconnaissance parfaite pour un observateur humain. Cette méthode peut être ajustée par l'opacité max du bruit afin de conserver une qualité optimale contre une réduction de l'efficacité contre les CNN.



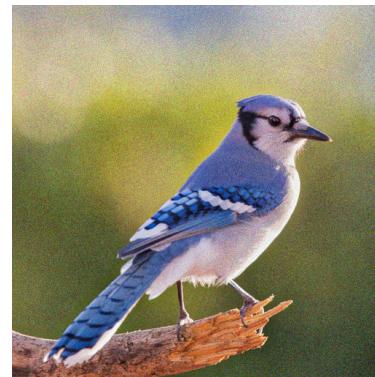
Opacité max = 65%

Précision CNN : ~10% (nulle)



Opacité max = 40%

Précision CNN : ~25%



Opacité max = 50%

Précision CNN : ~50%

Nous remarquons que certaines images, due à leur nature plus bruitée, reçoivent moins d'impacte visuel que d'autres. Cependant, pour toutes nos images, ce filtre réduit drastiquement la précision du CNN.

Pistes infructueuses

Nous avons essayé d'introduire de nombreuses transformations non linéaires à l'opacité ou à la carte de fusion, cependant il est difficile d'améliorer l'impact visuel du filtre sans diminuer son efficacité. Nous avons également essayé d'injecter des basses et hautes fréquences provenant d'autres images de classes différentes, cependant nous n'avons pas encore obtenu de résultats assez intéressants pour pousser cette piste cette semaine. Pour finir, nous avons tenté de remplacer le bruit actuel par un bruit de Perlin afin de rendre les transitions de bruits plus naturelles à l'œil, mais l'introduction d'un bruit non uniforme réduit grandement l'impact sur la précision du CNN.

Conclusion

Cette semaine, nous avons utilisé les résultats de nos approches naïves afin d'obtenir un filtre empêchant totalement la reconnaissance de notre CNN sans impacter la reconnaissance d'un observateur humain. Cependant cette méthode possède un impact visuel non négligeable sur les images et peut être facilement repérer par une analyse entropique de l'image (due au bruit introduit).

Annexes

Dépot Github

- <https://github.com/Pataleon/Securite-visuelle>