

Rapport du TP1 en Vision Artificielle et Traitement d'Images

ATTE Paul-Emmanuel - BERTRAND Hugo - EVORA Silva Frédérique - TOUGMA Boris

Introduction

Dans le cadre du TP1 en Vision Artificielle et Traitement d'Images, l'objectif de ce travail pratique est de développer un algorithme de détection de changements dans des images de différentes pièces (cuisine, salon et chambre) d'un appartement. Cet algorithme a pour but de mettre en évidence les changements, en mettant particulièrement l'accent sur les objets au sol, à l'aide de bounding boxes. Cette technologie pourrait s'avérer précieuse pour les personnes âgées vivant seules, en les aidant à éviter les accidents liés à la négligence de leur domicile.

Mécanisme de fonctionnement

Notre approche pour répondre au problème est la suivante:

Utilisation de script et algorithme: Le script `run.py` que nous avons développé parcourt tous les dossiers et fichiers à partir de l'adresse d'un dossier initial. Il applique l'algorithme de détection d'objet `algorithme.py` à tous les fichiers trouvés.

Gestion de l'image de référence : L'algorithme suppose qu'une image de référence est située dans le même dossier que les images à traiter. Si elle est absente, aucun traitement n'est effectué.

Mode d'emploi du scripte : Le scripte python final `run.py` prends en argument l'adresse d'un dossier qui contient des images et une image de référence.

Par exemple, le dossier `Image` ou le dossier `Image\Cuisine` pourraient être passés en argument.

Exemple d'exécution:

```
PS C:\Users\paule\OneDrive\Me\UQAC 1\8INF804 - Vision Artificielle\TP1_Vision> python run.py "C:\Users\paule\Desktop\Images"
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6567.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6568.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6569.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6570.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6571.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6572.JPG
Procesing file: C:\Users\paule\Desktop\Images\Chambre\IMG_6573.JPG
Procesing file: C:\Users\paule\Desktop\Images\Cuisine\IMG_6562.JPG
Procesing file: C:\Users\paule\Desktop\Images\Cuisine\IMG_6563.JPG
Procesing file: C:\Users\paule\Desktop\Images\Cuisine\IMG_6564.JPG
Procesing file: C:\Users\paule\Desktop\Images\Cuisine\IMG_6565.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6551.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6552.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6553.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6554.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6555.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6556.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6557.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6558.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6559.JPG
Procesing file: C:\Users\paule\Desktop\Images\Salon\IMG_6560.JPG
```

Toute l'arborescence du dossier **Images** est traitée.

Après traitement:

```
PS C:\Users\paule\Desktop\Images\Chambre> ls

Directory: C:\Users\paule\Desktop\Images\Chambre

Mode                LastWriteTime         Length Name
----                -
-a----          29-Jan-21  02:28 PM          8818990 IMG_6567.JPG
-a----          25-Oct-23  04:54 PM          6478590 IMG_6567.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          9160166 IMG_6568.JPG
-a----          25-Oct-23  04:54 PM          6741260 IMG_6568.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          9181396 IMG_6569.JPG
-a----          25-Oct-23  04:54 PM          6758889 IMG_6569.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          9201090 IMG_6570.JPG
-a----          25-Oct-23  04:54 PM          6786323 IMG_6570.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          8712596 IMG_6571.JPG
-a----          25-Oct-23  04:54 PM          6616483 IMG_6571.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          8360226 IMG_6572.JPG
-a----          25-Oct-23  04:54 PM          6337515 IMG_6572.JPG_objects_detected.jpg
-a----          29-Jan-21  02:29 PM          8614287 IMG_6573.JPG
-a----          25-Oct-23  04:54 PM          6405439 IMG_6573.JPG_objects_detected.jpg
-a----          29-Jan-21  02:28 PM          8896907 Reference.JPG
```

Après le traitement, les images modifiées sont sauvegardées.

Méthodes employées pour réaliser le TP

La plus grande leçon à tirer de ce TP est l'ensemble des méthodes et bonnes pratiques employées afin de répondre au problème qui a été posé:

Entraînement basé sur le machine learning : Principalement, la méthode que nous employons pour entraîner le modèle se base sur l'**entraînement d'un modèle d'apprentissage automatique**.

Nous commençons avec une idée générale de l'algorithme, nous établissons une métrique de performance afin d'évaluer comment est-ce que l'algorithme performe face au dataset.

Cette métrique nous sert de “**cost function**”, en fonction de cette valeur nous **ajustons les paramètres de l'algorithme afin d'obtenir de meilleures performances**. Cette méthode nous semble intéressante mais elle est **peu pratique** lorsque l'évaluation se fait de façon manuelle.

Nous effectuons de même une évaluation **qualitative** ou nous ciblons les conditions qui mettent l'algorithme particulièrement en difficulté.

Visualisation des résultats: Afin d'éviter le **overfitting** de l'algorithme nous essayons de visualiser les résultats obtenus par l'algorithme sur **l'ensemble** des images qui se trouvent dans le dossier Images. Cela nous permet d'avoir une idée générale de la performance de l'algorithme et de mieux discerner les conditions qui le mettent en difficulté.

Algorithme de détection pour les 3 pièces différentes

Notre algorithme de détection des changements comporte les étapes suivantes :

Conversion en niveaux de gris : Nous avons commencé par convertir les images en niveaux de gris pour simplifier le traitement et garder les informations utiles.

Calcul de la différence absolue : Pour détecter les changements, nous avons calculé l'image qui équivaut la différence absolue entre l'image traitée et l'image de référence.

Application des filtres : Pour réduire le bruit dans l'arrière-plan tout en préservant les objets à l'avant-plan, nous avons appliqué des filtres, notamment un filtre gaussien et un filtre d'ouverture.

Application du masque binaire : Un masque binaire a été appliqué pour séparer les objets de l'arrière-plan de ceux de l'avant-plan.

Détection des contours : En utilisant la méthode `findContours` de OpenCV, nous avons identifié les contours des objets qui se trouvent dans l'image après application du masque.

Dessin de bounding boxes : Nous avons ensuite dessiné des bounding boxes autour des objets qui tombent dans une plage d'aire spécifique. Nous évitons les bounding boxes des objets trop petits ou trop grands qui peuvent représenter des erreurs de l'algorithme.

Entraînement du modèle

Pour entraîner notre algorithme nous avons suivi les étapes suivantes :

Application du script à des images individuelles : Nous avons appliqué l'algorithme à des images individuelles de manière aléatoire pour observer la performance de l'algorithme sur différentes situations. Lorsque nous trouvons que l'algorithme présente des résultats satisfaisants face à différentes images, nous appliquons le script qui permet de traiter l'ensemble des images qui se trouvent dans le dossier images.

Analyse quantitative et qualitative : Pour évaluer les performances de l'algorithme, nous avons réalisé une analyse quantitative en comptant le nombre d'images avec une **détection bonne, moyenne et mauvaise**. De plus, une analyse qualitative a été effectuée pour identifier les **conditions d'image** où l'algorithme a eu plus de difficultés.

Ajustement de l'algorithme : L'algorithme a été ajusté en fonction des résultats de l'analyse, afin d'améliorer ses performances.

Répétition du processus : Nous avons répété ces étapes jusqu'à obtenir des résultats satisfaisants.

Résultats obtenus

Dans l'ensemble nous trouvons que l'algorithme présente des **résultats satisfaisants**, en général les objets sur le sol sont détectés mais avec quelques objets de l'arrière-plan.

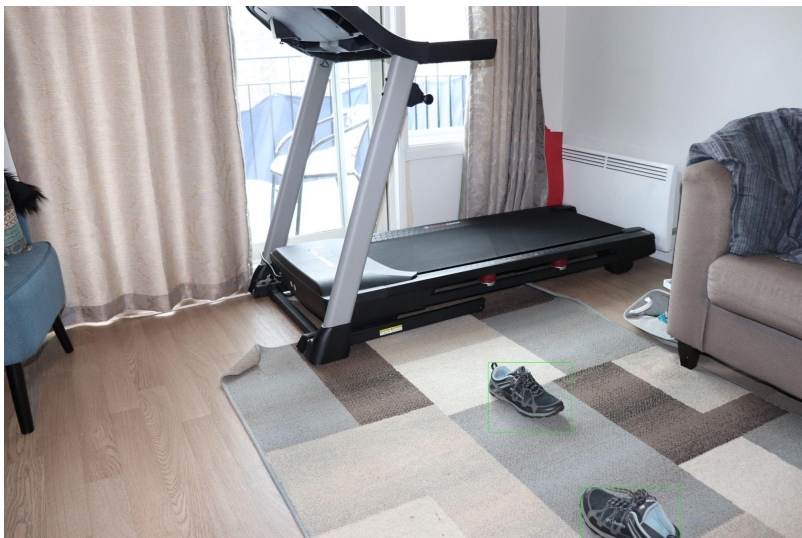
La difficulté principale de l'algorithme semble être les photos prises avec un **décalage de la position de la caméra** et les **photos sombres ou trop éclairées**. L'algorithme pourrait être nettement amélioré avec l'utilisation d'un **modèle de machine learning déjà entraîné** pour la détection d'objets.

Exemple de bonne détection : Nous trouvons que la détection est “bonne” lorsque les principaux objets sur le sol sont détectés avec peu d'objet de l'arrière plan entouré.

Référence:



Image traitée:



L'algorithme détecte bien les deux chaussures sur le sol, aucun objet de l'arrière-plan est entouré.

ATTE, BERTRAND, EVORA, TOUGMA,

Exemple de détection moyenne : Nous considérons que la détection est moyenne lorsque les principaux objets sur le sol sont entourés et il y a autant d'objets de l'arrière-plan détectés que ceux de l'avant-plan.

Référence:



Image traitée:



Dans ce cas les trois objets au sol sont bien détectés, mais l'algorithme détecte également des objets sur le carton qui étaient présents dans l'image de référence.

ATTE, BERTRAND, EVORA, TOUGMA,

Exemple de mauvaise détection : Nous considérons que la détection est moyenne lorsque l'algorithme ne détecte pas les objets au sol et une grande quantité d'objets de l'arrière-plan sont détectés. En général, nous constatons que cela apparaît lorsque **l'angle de la caméra est modifié** et **les conditions de luminosité sont assez différentes** entre l'image de référence et l'image traitée.

Référence :



Image traitée :



Aucun objet sur le sol n'est détecté et une grande quantité des objets de l'arrière-plan sont entourés.

ATTE, BERTRAND, EVORA, TOUGMA,

Difficultés rencontrées

Analyse manuelle de la performance du script : L'analyse des performances de l'algorithme a nécessité une évaluation manuelle, ce qui a été une tâche intensive. Tandis que le traitement d'image se fait assez rapidement, après avoir modifié l'algorithme cinq ou six fois, les résultats obtenus semblent peu changés à l'œil nu.

Évitement du "overfitting" de l'algorithme : Nous avons dû prendre des mesures pour éviter que l'algorithme ne s'ajuste trop aux données d'entraînement, ce qui aurait pu le rendre moins généralisable. Lors du début du travail, des essais pour corriger des erreurs sur certaines photos ont conduit à des problèmes sur d'autres. Ainsi une analyse globale des images semblait plus efficace.

Ajustement des paramètres des filtres : Les paramètres des filtres ont dû être ajustés en fonction de l'analyse qualitative et quantitative des résultats pour optimiser les performances de l'algorithme. Certains paramètres plus techniques tel que la valeur sigma du kernel ont nécessité plus de recherches.

Conclusion

Ce TP nous a permis de développer un algorithme de détection de changements d'une grande utilité potentielle, en particulier **pour** les personnes âgées vivant seules. La leçon principale tirée de ce TP est **qu'aucune solution n'est parfaite**, on souhaite avoir un algorithme qui a des **résultats satisfaisants sur l'ensemble d'un dataset** et non qui fonctionne parfaitement sur une seule image.

Le critère essentiel de performance de l'algorithme est sa **capacité à avoir des performances acceptables sur différentes situations**.

Améliorations possibles:

1. Soigner les paramètres des filtres pour obtenir des meilleurs résultats pour les conditions d'éclairage variées
2. Utilisation d'un réseau neuronal déjà entraîné
3. Modifier `run.py` pour qu'il puisse être exécuté sur un seul fichier, au lieu d'uniquement sur l'adresse d'un dossier

Le script que nous avons élaboré est capable de **traiter des images de différentes pièces, détecter les changements au sol** et mettre en valeur ces changements à l'aide de **bounding boxes**. Nous avons également acquis de précieuses compétences en **ajustant l'algorithme en fonction des performances** et en gérant les défis liés à l'analyse des résultats. Ce travail constitue une première étape vers la création d'un système bénéfique pour de nombreux utilisateurs.

ATTE, BERTRAND, EVORA, TOUGMA,