



جامعة دمشق

كلية الهندسة المعلوماتية

الرؤية الحاسوبية

الوظيفة الثانية (Morphological Operations)

تقدمة:

أسامه يوسف بازو

ياسين راتب عبد المهدي

26-10-2023

القسم الأول

قمنا بإنشاء Trackbar بجانب النوافذ التي تعرض الصور وذلك لسهولة التعديل على القيم من قبل المستخدم.

قمنا في البداية بتعرف المتحولات المطلوبة والتي تلعب دورا في تأثير الفلاتر على الصور التي يتم تعديلها والتي سنقوم بالتعديل عليها من خلال الـ trackbar الذي ذكرناه سابقا وهي:

- gaussian_blur_kernel_size قيمة النافذة التي تقوم بتطبيق تأثير غاوس على الصورة.
- median_blur_kernel_size قيمة النافذة التي تقوم بتطبيق تأثير الوسيط على الصورة.
- canny_threshold_low و canny_threshold_high والتي تعبر عن حد العتبة العليا والدنيا لتأثير كاني على الصور.
- erosion_iterations والذي يعبر عن عدد الدورات التي تتم لمعالجة الصورة من خلال الحت.
- dilation_iterations والذي يعبر عن عدد الدورات التي تتم لمعالجة الصورة من خلال التمديد.
- opening_iterations والذي يعبر عن عدد الدورات التي تتم لمعالجة الصورة من خلال الفتح.
- closing_iterations والذي يعبر عن عدد الدورات التي تتم لمعالجة الصورة من خلال الإغلاق.

ومن ثم قمنا بإنشاء التوابع التي تقوم بالتعديل على قيم المتحولات السابقة لرؤية الفرق في التأثير على صورة الدخل التي تتم معالجتها.

لتأتي بعدها التوابع التي تقوم بالقيام بالعمليات المورفولوجية وتطبيقها على صورة الدخل.

قمنا بمحاولة معالجة الصورة التالية:



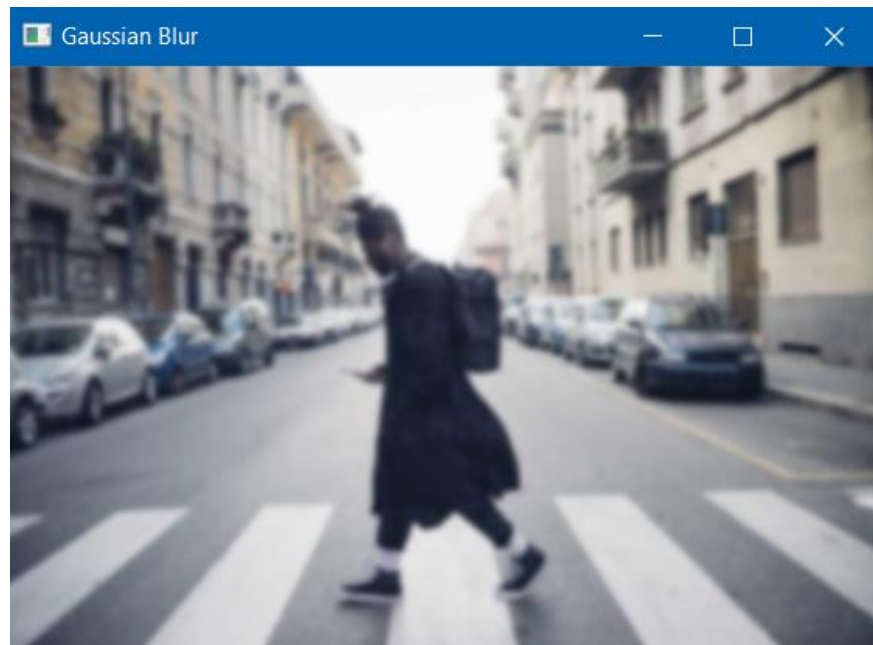
حيث كانت الفكرة هي محاولة معرفة في حال كان أحد المشاة يقوم بعبور الطريق وذلك بافتراض وجود كاميرا تقوم بالمراقبة، تتم معالجة الفكرة السابقة من خلال التعرف على الحواف في الصورة من خلال تأثير canny حيث نقوم بتطبيق تأثير GaussianBlur على الصورة قبل تطبيق تأثير canny عليها لتنتج الصورة التالية:



للحصول على الصورة السابقة نقوم بتعيين العتبات الدنيا والسفلى لتأثير canny لأقصى قيمة ممكنة ويتم تعيين قيمة الـ kernel الخاصة بـ gaussianBlur لتكون قيمتها 9.

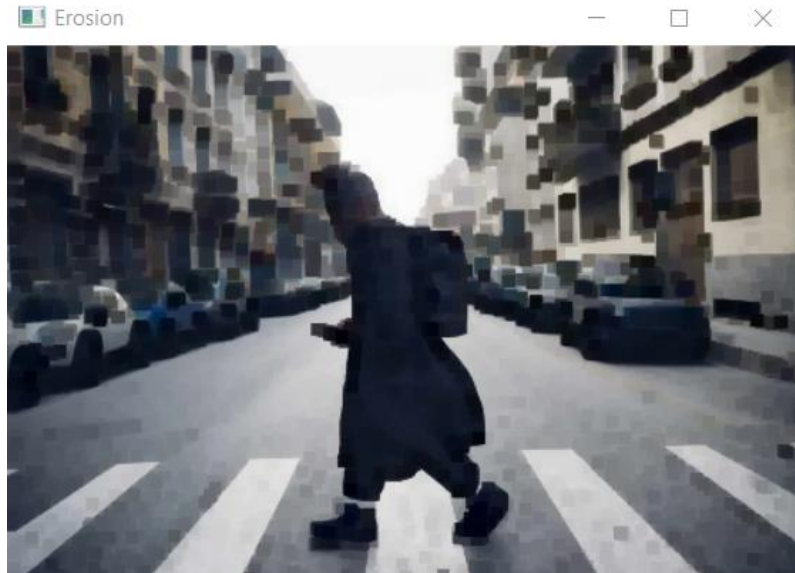
من الصورة السابقة نلاحظ وجود بعض الحواف التي بقيت والتي من الممكن ازالتها وذلك بتطبيق تأثير الحت.

حيث تكون الصورة الأصلية بالشكل:



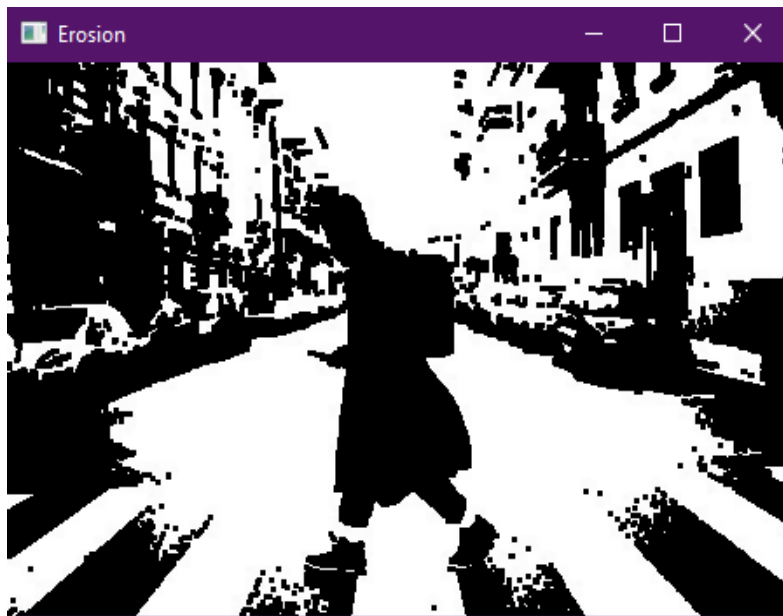
قمنا بتطبيق العمليات المورفولوجية على الصورة لمعرفة مدى تأثير هذه العمليات على الصورة والاستفادة منها فكانت النتائج كما يلي:

عند تطبيق عملية الحت على الصورة الاصلية كانت النتائج كما يلي وذلك بوضع قيمة المعامل لتكون 6:



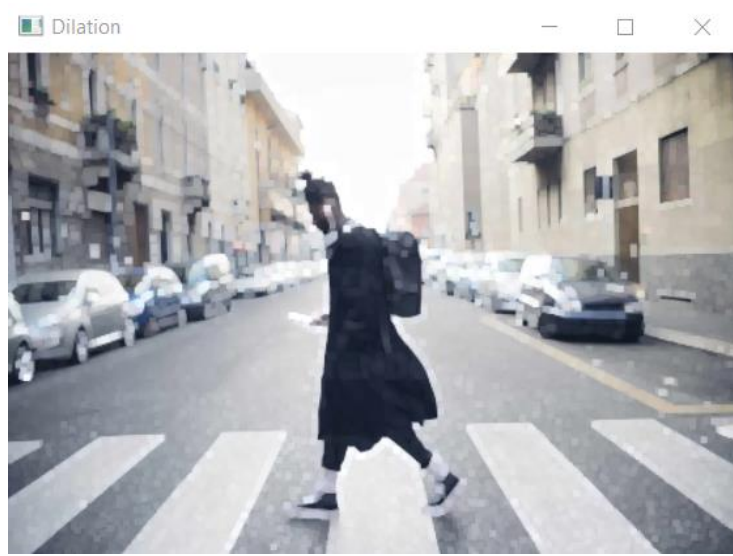
نلاحظ ان الحت قد يؤدي الى زيادة الحدة في الحواف كما نرى في ممر المشاة بالإضافة إلى ظهور بعض المربعات التي كانت غير موجودة مسبقاً.

وعندما طبقنا الحت بعد تحويل الصورة ل Binary تكون النتيجة كالتالي:



اي نلاحظ ازدياد في الحواف ذات اللون الاسود ولكن عند زيادة الـ iterations أكثر من ذلك ستصبح الصورة غير مفهومة وسيصبح من الصعب التخلص من التفاصيل الناتجة عن هذه العملية كما نرى في الصورة الثانية.

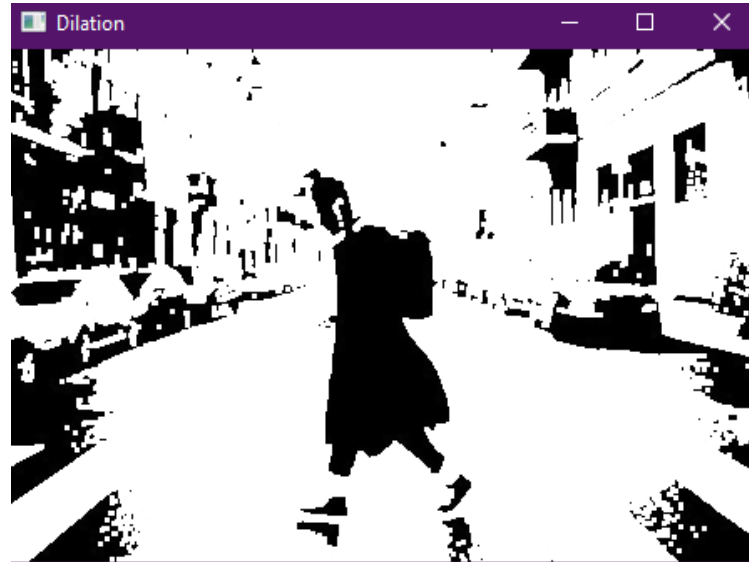
أما عند تطبيق التمديد بوضع المعامل ليكون 3 :



نجد أن التمديد في حالتنا ذو فائدة أكبر حيث قام بطمس بعض المعالم من السيارات في الخلف مع المحافظة على حواف الجسم للرجل الذي يعبر الطريق.

فيمكن في هذه الحالة أن نقوم بتطبيق التمديد وفق المعامل السابق ومن ثم تطبيق تأثير canny للحصول على نتائج أفضل بما يخص حواف المارين من الطريق.

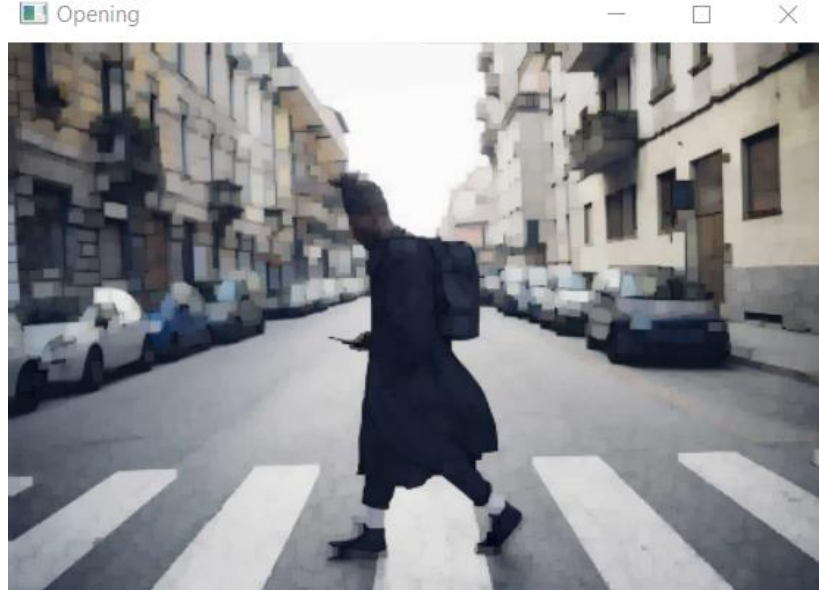
وايضاً عند تطبيق التمديد على صورة Binary نلاحظ النتيجة التالية:



اصبحت معالم الشخص هنا واضحة بعض الشيء والنتيجة أفضل من الحت حيث نرى فرق النتيجة عن الحت من ناحية عدم زيادة التفاصيل كما نتج في عملية الحت.

قمنا بتطبيق أخيرا تأثيري الفتح والاعلاق وسنقوم بمناقشة النتائج:

بما يخص الفتح كانت النتيجة عند وضع المعامل بالقيمة 4 كالتالي:



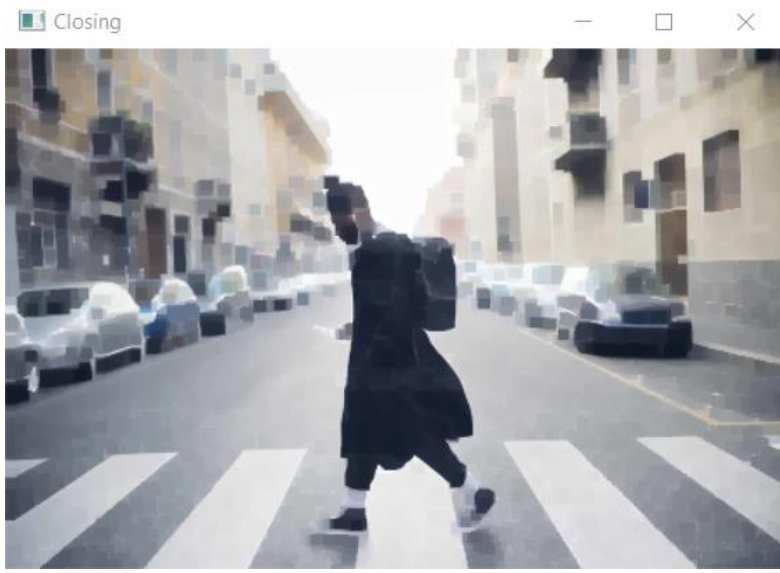
نستطيع ملاحظة الفرق عن عملية الحت السابقة وذلك بأن حواف ممر المشاة لم تزداد ولم تظهر المربعات فيها كما حدث في عند تطبيق عملية الحت.

وعند التطبيق على صورة Binary نلاحظ النتيجة التالية:



نجد أنه عند تطبيق الفتح أن الحواف من المنطقة العلوية أقرب لأن تندمج مع المحيط مما قد يؤدي لخسارة في الحواف في المنطقة العلوية فيفضل في هذه الحالة عدم استخدام عملية الفتح.

بما يخص الفتح كانت النتيجة عند وضع المعامل بالقيمة 5 كالتالي:



عند تطبيق عملية الإغلاق نجد أنها أفضل ما يمكن من ناحية الحفاظ على المحيط الخارجي لجسم المار من الطريق بالإضافة الى عدم زيادة التفاصيل بشكل قد يؤثر سلباً، فنجد مربعات سوداء قريبة من منطقة الرأس ولكن في هذه الحالة ف من الممكن اعتبار هذا النوع من الزيادة مقبولا لعدم الزيادة في الشكل او تمديده ليصبح خارجا عن المألوف.

أخيراً قد طبقنا عملية ال Dilation بشكل يدوي:

```
1 def manual_dilation(binary_image, kernel):
2     height, width = binary_image.shape
3     k_height, k_width = kernel.shape
4     dilated_image = np.zeros((height, width), dtype=np.uint8)
5     for x in range(k_height // 2, height - k_height // 2):
6         for y in range(k_width // 2, width - k_width // 2):
7             if binary_image[x, y] == 255:
8                 dilated_image[x - k_height // 2: x +
9                             x + k_height // 2 + 1, y - k_width // 2:
10                            y + k_width // 2 + 1] = 255
11     return dilated_image
12
13
14 binary_image = cv2.imread('binary.jpg', 0)
15 kernel = np.array([[0, 255, 0],
16                    [255, 255, 255],
17                    [0, 255, 0]], dtype=np.uint8)
18 dilated_image = manual_dilation(binary_image, kernel)
19 cv2.imwrite('manually_dilated_image.jpg', dilated_image)
```

بأخذ التابع بارامترين الصورة Binary وإيضاً ال Kernal حيث نأخذ بالبداية طول وعرض الصورة وإيضاً طول وعرض ال kernel وبعد ذلك تكون هنالك الصورة الناتجة وهي dilated_image حيث بالبداية تكون سوداء كل البكسلات قيمتها صفر وبعد ذلك ننشئ حلقة على الصورة الباينري عن طريق حجم ال kernel ونرى إذا كان البكسل لونه أبيض من الصورة الباينري عندها يجب ان يكون هذا البكسل أبيض في الصورة الناتجة وهكذا.

الخلاصة :

بعد التجريب والاختبار يمكن استخلاص أن تطبيق عملية كاني بدون تطبيق أي نوع من الفالتر من الممكن أن تكون ذو فائدة وتعطي نتائج دقيقة، ولكن من الممكن أن نقوم بتطبيق بعض الفالتر مثل تطبيق عملية الإغلاق أو تطبيق فلتر التمديد لتحسين جودة النتائج واستخراج الحواف بشكل أفضل.

القسم الثاني

أخيراً سنأخذ الصور الناتجة عن العمليات Erosion, Dilation, Closing, opening ونطبق العمليات المنطقية AND, OR, XOR مع الصورة الاساسية:

أولاً عند تطبيقها مع ال Erosion:

AND •



OR •



XOR •



عند تطبيقها مع ال Dilation:

AND •



OR •



XOR •



مع الد Closing:

AND •



OR •



XOR •



مع ال Opening:

AND •



OR •



XOR •

