# Improving the Performance of Medical Image Processing Systems Using Graphics Processing Units (GPUs)

**Dr. Ya,aroub Bader***
**Ahmad Hasan****

**(Received 10 / 11 / 2024. Accepted 26 / 1 / 2025)**

## □ ABSTRACT □

Medical imaging currently plays a crucial role in all clinical applications of medical procedures, ranging from medical research to diagnosis and treatment planning. However, medical imaging procedures are often computationally demanding due to the large 3D medical datasets processed in practical clinical applications.

Thanks to the rapid enhanced performance of graphics processing units (GPUs), improved programming support, and their excellent cost-to-performance ratio, the GPU has emerged as a viable parallel computing platform for computationally intensive tasks requiring high processing power.

This study aims to leverage distributed computing techniques on the GPU to enhance and accelerate the processing of medical images, which could play a pivotal role in long-term therapeutic procedures. The algorithm was developed and applied in three key areas of medical image processing: segmentation, filtering, and edge detection.

**Keywords**: Central Processing Unit (CPU), Graphics Processing Unit (GPU), Image Segmentation, High-Performance Computing, Parallel Processing, CUDA.

**Copyright** **:**Tishreen University journal-Syria, The authors retain the copyright under a CC BY-NC-SA 04

* **Assistant Professor - Department of Automatic Control Engineering, Faculty of Mechanical and Electrical Engineering, Latakia University, Lattakia, Syria.**
****MSc. Student - Department of Automatic Control Engineering, Faculty of Mechanical and Electrical Engineering, Latakia- University, Lattakia, Syria.**

# تحسين أداء أنظمة معالجة الصور الطبية باستخدام وحدات المعالجة الرسومية

**د. يعرب بدر***

**أحمد حسن****

**(تاريخ الإيداع 10 / 11 / 2024. قُبِل للنشر في 26/ 1 / 2025)**

## □ ملخّص □

يلعب التصوير الطبي حالياً دوراً حاسماً في جميع التطبيقات السريرية للمعالجات الطبية بدءاً من البحث العلمي الطبي وحتى التشخيص وتخطيط العلاج. ومع ذلك، غالباً ما تعتبر إجراءات التصوير الطبي متطلبة حسابياً نظراً لمجموعات البيانات الطبية الكبيرة ثلاثية الأبعاد (3D) والتي يتم معالجتها في التطبيقات السريرية العملية.

بفضل الأداء المعزز السريع لمعالجات الرسوميات، ودعم البرمجة المحسن، ونسبة الكلفة إلى الأداء الممتازة والتي تتمتع بها، برزت وحدة معالجة الرسوميات (GPU) كمنصة حوسبة متوازية قابلة للاستخدام في المهام المكلفة حسابياً ذات الحاجة إلى قدرات معالجة عالية.

تهدف الدراسة إلى استخدام تقنيات الحوسبة الموزعة على وحدة معالجة الرسوميات GPU من أجل تحسين وتسريع معالجة الصور الطبية والتي قد تلعب دوراً حاسماً في العمليات العلاجية على المدى البعيد. تم تطوير وتطبيق الخوارزمية للعمل في ثلاث مجالات لمعالجة الصور الطبية، وهي عمليات التجزئة والترشيح وكشف الحواف.

**الكلمات المفتاحية:** وحدة المعالجة المركزية (CPU)، وحدة المعالجة الرسومية (GPU)، تجزئة الصور، الحوسبة عالية الأداء، المعالجة المتوازية، CUDA.

**حقوق النشر** : مجلة جامعة تشرين- سورية، يحتفظ المؤلفون بحقوق النشر بموجب الترخيص CC BY-NC-SA 04

* **مدرس – قسم هندسة الحاسبات والتحكم الآلي – كلية الهندسة الميكانيكية والكهربائية – جامعة اللاذقية- اللاذقية- سورية.**

** **طالب دراسات (ماجستير) – قسم هندسة الحاسبات والتحكم الآلي –كلية الهندسة الميكانيكية والكهربائية – جامعة اللاذقية – اللاذقية- سورية.**

## مقدمة:

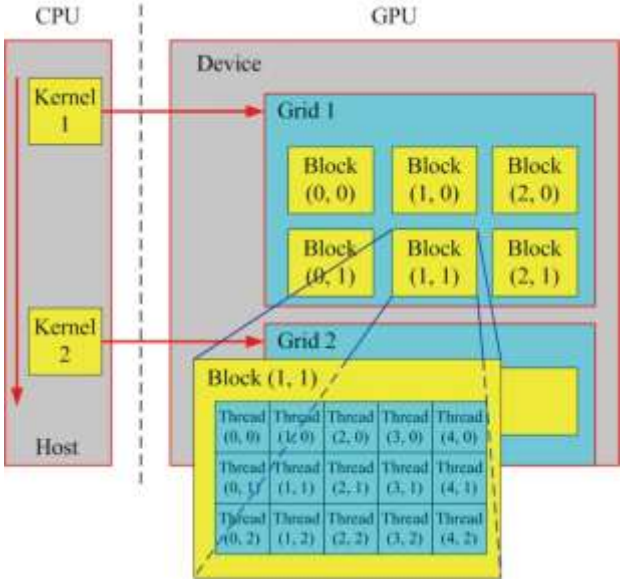
تُمثل الحوسبة المتوازية مستقبل الحوسبة عالية الأداء في التطبيقات الطبية. على مدى السنوات العديدة الماضية، تطورت وحدة معالجة الرسوميات كمنصة تسريع عالية الأداء للحوسبة المتوازية، خصوصاً في معالجة الصور الطبية وتحليلها. نظراً للطلب المتزايد على الحوسبة عالية الأداء في التطبيقات الرسومية والعلمية المتقدمة، تطورت أجهزة الرسوميات التجارية من بنية قابلة للبرمجة إلى بنية شبيهة بالحواسيب الفائقة القابلة للبرمجة. في الوقت ذاته، تطورت وحدة معالجة الرسوميات لتصبح منصة أكثر فعالية مع الحفاظ على نسبة ممتازة بين السعر والأداء لمجموعة واسعة من المهام الحسابية [1].

صُممت وحدة معالجة الرسوميات لتلائم فئات معينة من التطبيقات التي تتطلب قدرة حسابية عالية، ودرجة عالية من التوازي، وزمن معالجة سريع مع تقليل زمن الوصول لأدنى حد ممكن [4].

في السنوات الأخيرة، زادت السرعة الحسابية لوحدة معالجة الرسوميات بشكل ملحوظ، مما يوفر تسريعاً كبيراً للمهام ذات العبء الحسابي مقارنةً بإطار الحوسبة التقليدي الذي يعتمد على وحدة المعالجة المركزية. ورغم ذلك، فإن وحدة معالجة الرسوميات غير مناسبة للأغراض العامة أو الحسابات غير الرسومية. تُعرف الجهود المبذولة لاستخدام وحدات معالجة الرسوميات في الحوسبة العامة بـ "الحوسبة للأغراض العامة على وحدة معالجة الرسوميات" (GPGPU) [5].

تُجرى حسابات GPGPU باستخدام معالجات الرسوميات المتخصصة بدلاً من المعالجات التقليدية، مما يتطلب خبرة في واجهات برمجة التطبيقات الرسومية للاستفادة من إمكانيات منصة GPU/GPGPU التقليدية، وهو ما قد يشكل تحدياً للمستخدمين غير المتخصصين. لحسن الحظ، أدى ظهور تقنية Compute Unified Device Architecture (CUDA) من NVIDIA في أواخر عام 2006 إلى التغلب على هذه التحديات. توفر CUDA بيئة تطوير شبيهة بلغة C، مما يسهل البرمجة العامة المتوازية على وحدات معالجة الرسوميات من NVIDIA، ويتيح للمطورين والباحثين استغلال القدرات الهائلة لهذه الوحدات في مجموعة متنوعة من التطبيقات الحسابية[2] [5] [10] .

يوضح الشكل 1 نظرة عامة تخطيطية لنموذج برمجة CUDA.




**الشكل (1) بنية مخطط نموذج برمجة CUDA**

## أهمية البحث وأهدافه:

يهدف البحث إلى إجراء عمليات معالجة الصور الطبية على وحدات المعالجة الرسومية لتحقيق التحليل الطبي في الوقت الحقيقي. تتميز وحدات المعالجة الرسومية بقدرتها على تنفيذ العديد من العمليات، حيث يتم تخصيص وحدة معالجة مستقلة لكل جزء من الصورة، مما يتيح تسريع عمليات المعالجة بشكل كبير، ويُسهم في التنبؤ بالأمراض وكشف المناطق المصابة.

تسمح هذه الدراسة بتوجيه الأطباء ومراكز التحليل إلى استخدام الوحدات الرسومية لتجنب فترات الانتظار الطويلة في معالجة الصور التي يتم الحصول عليها من أجل كشف الحالات المرضية.

## طرائق البحث ومواده:

يعتمد البحث على توزيع المهام على وحدات GPU حيث تُقسَّم الصورة إلى مقاطع تُعالج كل منها بشكل مستقل قبل إعادة تجميعها. تمت مقارنة زمن المعالجة التسلسلية بزمن المعالجة التفرعية، مما أظهر بوضوح الفارق بين المعالجة التسلسلية والمعالجة التفرعية.

اعتمدت الدراسة على وحدة GPU من نوع Nvidia GeForce 930MX والمقارنة مع أداء وحدة CPU من نوع Core i7-6400MQ.

**الدراسات المرجعية:**

على الرغم من زيادة الأبحاث حول تحسين أنظمة تجزئة الصور واستخدامها المتزايد في العقود الأخيرة، إلا أنها ليست مناسبة لتطبيقات الزمن الحقيقي بسبب وقتها الحسابي الكبير. ولحسن الحظ، يمكن تنفيذ أساليب التجزئة في بيئة تشغيلية في الزمن الحقيقي من خلال تكييف هذه الأساليب مع بنيات وحدة معالجة الرسوميات[12] [1] [5] [14].

يستعرض الجدول 1 بعض أساليب معالجة الصور المستندة إلى وحدة معالجة الرسوميات وخصائص كل منها.

**الجدول 1 بعض أساليب معالجة الصورة المستندة إلى وحدة معالجة الرسوميات**

| النوع | البحث | التطبيقات | الخصائص |
|---|---|---|---|
| 2D CT MRI | نموذج تحليل المحيط الفعال | الدماغ | غير مثالي لتطبيقات الصور الطبية بسبب اقتصاره على تقسيم المنطقة إلى قسمين فقط. |
| 3D CT | التعليم الفعال | منطقة الحوض | تقليل إدخالات المستخدمين للصور في مهام تجزئة الصور التفاعلية ثلاثية الأبعاد. |
| 3D CT MRI | على مجال المستويات المختلفة | على مستوى الكلية والدماغ | خوارزمية التجزئة الأولى والوحيدة على مستوى وحدة معالجة الرسومات مع العمل على مستوى التعقيد الخطي – والتعقيد اللوغاريتمي. |

| | | | |
|---|---|---|---|
| | | | |
| 3D CT MRI | تقنية النقطة المشعة | الدماغ | إعطاء معلومات عالية الجودة في الوقت الفعلي للمناطق المجزأة |
| 3D MRI | مجموعة البيانات على مستوى الأسراب | الدماغ | توفر مجموعة بيانات الأسراب بيئة مثالية عالية الأداء ومقاومة للضجيج |
| 3D MRI | تقنية هجينة | الدماغ | تقنية تجزئة هجينة تفاعلية تجمع بين عمليات توسيع المناطق القائمة على التعتيب والانتشار. |
| 3D MRI | تحديد المناطق باستخدام البذرة Seed | الدماغ والهيكل العظمي | يستخدم بسهولة في عدد من التطبيقات بما في ذلك الأنظمة القائمة على تحليل النقاط الأخرى المجاورة والشبكات متعددة الأضلاع والحجم غير المنتظم مع الهيكل المتغير |
| 3D MRI | التحليل الخلوي الآلي | الكلى | من التحاليل البسيطة والمناسب والفعال في المجال الطبي |

تم تنفيذ أول تطبيق لتقنية تجزئة الصور الطبية المعتمدة على وحدة معالجة الرسوميات (GPU) عبر صياغة عملية تجزئة مجموعة المستويات كسلسلة من أجهزة تشغيل الرسوميات لمزج الصور. حققت النماذج القابلة للبرمجة التي تم تقديمها قدراً أكبر من المرونة ومكّنت من تجزئة الصور ثلاثية الأبعاد باستخدام تنظيم التحكم بالانحناءات لعرضها على شكل أسطح متساوية وملساء. تم لاحقاً تنفيذ التقنية على شكل مجموعة من المستويات الجديدة، مع تقسيم العمل بطريقة مشابهة للعمليات العصبية وتوزيعه على وحدات CUDA [15].

يمكن تصنيف طرق التجزئة بشكل عام إلى فئتين رئيسيتين: التحليل منخفض المستوى والتحليل عالي المستوى. تتعامل الأساليب منخفضة المستوى، التي لا تتطلب معلومات إحصائية حول أنواع الكائنات الموجودة في الصورة، بشكل مباشر مع معلومات البكسل لتشكيل كائنات/مناطق ذات أهمية متصلة [12] [15] .

تمّ تقديم إطار عمل يعتمد على التجزئة باستخدام المعالجة المتوازية في بيئة CUDA من NVIDIA، لتجزئة الصور الفراغية باستخدام نماذج قابلة للتعديل. كما تم اقتراح تطبيقات مختلفة لأساليب تجزئة الصور الطبية باستخدام وحدات معالجة الرسوميات التي تدعمCUDA، مما حقق أداءً عالياً من حيث التسريع مقارنةً بالإصدار التسلسلي من الخوارزميات. تستمر الصور الطبية في الزيادة من حيث الحجم، مما يزيد الحاجة إلى المعالجة[14].

بالنسبة لتجزئة الصور عالية الأبعاد في التطبيقات السريرية في الزمن الحقيقي، من الضروري تنفيذ أساليب تجزئة الصور في الوقت الحقيقي لمجموعات البيانات الكبيرة باستخدام تقنية GPU/CUDA. تبقى التجزئة عالية الأبعاد المعتمدة على GPU/CUDA وذات التكلفة الحسابية المنخفضة تحدياً مستمراً للتصوير الطبي في المستقبل [18].

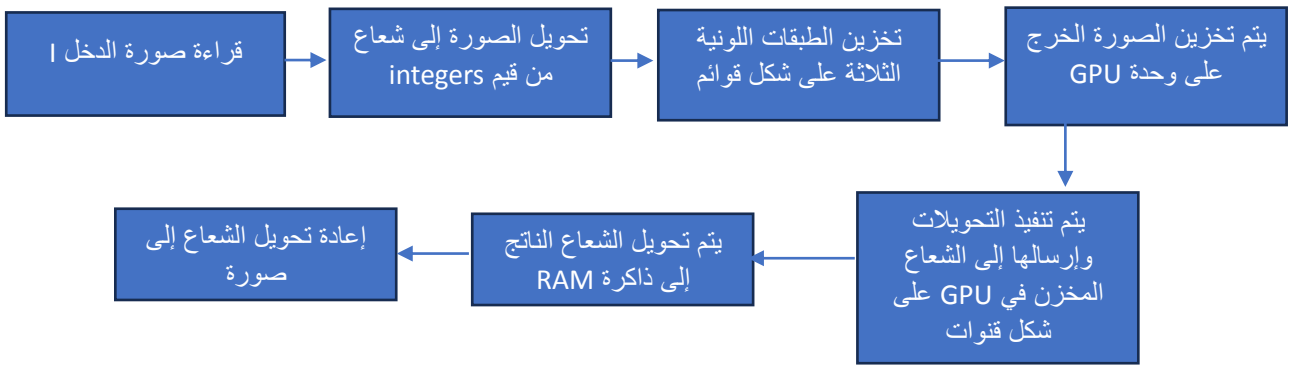
تعاني أساليب تجزئة مجموعة المستويات السابقة من كونها مكلفة حسابياً، حتى عند تشغيلها على وحدة معالجة الرسوميات. في إطار خوارزمية متوازية ذات كفاءة عالية، تم تقديم خوارزمية تجزئة مجموعة المستويات الجديدة

المعتمدة على وحدة معالجة الرسوميات، والتي أثبتت فعاليتها على جميع الأصعدة. حققت هذه الخوارزمية تقليصاً في مجموعة المستويات المعالجة بمقدار 16 مرة، وكانت أسرع بمقدار 14 مرة من الخوارزميات السابقة المستندة إلى وحدة معالجة الرسوميات مع الحفاظ على دقة التجزئة. مثلت هذه الطريقة الخوارزمية الأولى لتجزئة مجموعة المستويات على وحدة معالجة الرسوميات مع الأخذ بعين الاعتبار تعقيد العمل الخطي وتعقيد الخطوة اللوغاريتمية [18] [19].

### 1-1 الخوارزمية المقترحة:

تهدف الدراسة إلى توزيع عمليات معالجة الصورة الرقمية على وحدات معالج الرسوميات لتحقيق معالجة الصور الطبية في الوقت الحقيقي. تمت التجارب على معالج حاسوب من نوع Core i7 من الجيل السادس ووحدة معالج رسوميات متوسطة الأداء من نوع Nvidia Geforce 930M التي تحتوي على 384 نواة معالجة وتدعم تقنية CUDA. بالإضافة إلى ذلك، تم الاعتماد على نسخة Cuda11 التي تتيح البرمجة التفرعية على وحدة المعالجة الرسومية.

تعتمد الخوارزمية المقترحة على مجموعة من الخطوات الثابتة في المراحل الثلاث الأولى، بينما تتغير الخطوة الرابعة وفقاً للعملية المطلوبة. في البداية، تقوم وحدة المعالجة المركزية (CPU) بإعداد مجموعة من الكتل (Blocks) بحجم x3232، حيث يعالج كل منها مجموعة من الخيوط (threads) التي يمكن تعريفها بالشكل التالي: يتم تقسيم طول وعرض الصورة على 1024، وبالتالي تحتوي كل صورة بحجم x10241024 على 1024 خيطاً في كل كتلة. وقد تم اختيار هذا الرقم لأنه الحد الأقصى لعدد الخيوط التي تدعمها وحدة المعالجة الرسومية.




**الشكل (2) مخطط المعالجة في النظام المقترح**

بعد توزيع المهام، سيتم إرسال كل كتلة في حالة المعالجة التفرعية إلى قسم مخصص لمعالجتها. ونظراً لأن الصورة تحتوي على مجموعة من الكتل المستقلة، فلن تكون هناك علاقة بين أي قسم والأقسام الأخرى التي يتم معالجتها، أي أنه لا حاجة لانتظار أحد الأقسام حتى ينهي قسم آخر المهمة المسندة إليه. بعد انتهاء كل كتلة من عملية المعالجة الخاصة بها، ستعيد النتيجة إلى الذاكرة على شكل شعاع يتم ترتيبه وفقاً للتسلسل الذي أُرسلت به، ثم تُعاد تجميع هذه الكتل لتشكيل الصورة الأصلية.

في حالة المعالجة التسلسلية، تُنفذ كل كتلة بصورة مستقلة، ويجب انتظار الكتلة الحالية حتى تنتهي قبل بدء الكتلة التالية، مما يتطلب زمناً أكبر يزداد بشكل ملحوظ مع زيادة حجم الصورة. تم اعتماد أربعة أحجام لكل نموذج من نماذج الصور المدروسة وهي 128*128، 256*256، 512*512، و1024*1024، وتم إجراء كل عملية من العمليات عليها وحساب الزمن المطلوب للمعالجة.

تُحسب نسبة التسريع بقسمة مجموع الأزمنة أثناء المعالجة التسلسلية على مجموع الأزمنة أثناء المعالجة التفرعية، وذلك لأن بعض العمليات البسيطة استغرقت زمناً مهملاً (صفر مع الأخذ بعين الاعتبار 8 أرقام بعد الفاصلة).

### 2-1 سيناريوهات عملية الاختبار:

تم إجراء الاختبارات على مجموعة من الصور الرقمية من نمط MRI، والتي تتضمن درجات لونية مختلفة لصور الدماغ (مثل صور سرطان الدماغ) بالإضافة إلى صور شعاعية للصدر. تم إجراء هذه الاختبارات على العمليات التالية:

1. تجزئة السويات الرمادية: تتيح هذه العملية الوصول إلى مستويات مختلفة من التدرجات الرمادية لتوضيح أجزاء متعددة من الصورة، حيث يتم تحديد 8 مستويات رمادية تتراوح من المستوى المنخفض (الذي يتضمن المعلومات العامة في الصورة) إلى المستوى الأعلى الذي يمثل الحواف. في التطبيق المقترح، تُجرى هذه العملية عن طريق أخذ المتوسط الحسابي للطبقات الثلاث في حالة الصور الملونة وتحويلها إلى طبقة واحدة ذات تدرج رمادي.
2. إيجاد معكوس الصورة: (Negative Image) في هذه العملية، يُستخدم أعلى مستوى رمادي للصورة (وهو 255 في الصور ذات الدقة 8 بت) وتُطرح قيمة كل بكسل من هذه القيمة. تتيح عملية العكس هذه تشكيل قناع لكشف مناطق محددة في الصورة الطبية باستخدام العمليات المنطقية.
3. تعتيب الصورة: (Image thresholding) هي العملية التي تُحول فيها الصورة الرمادية إلى مستويين فقط؛ أدنى مستوى يمثَّل بالصفر (اللون الأسود) وأعلى مستوى يمثَّل بالرقم 1 في الصور الثنائية أو 255 في الصور ذات الدقة 8 بت. تتيح عملية تعتيب الصورة كشف وعزل المناطق المهمة لتوضيح مواقع هامة في الصورة.
4. زيادة السطوع: (Brightening) هي عملية يُرفع فيها معدل اللون الأبيض في الصورة إذا كانت الصورة عاتمة بشكل كبير. تتيح هذه العملية تمييز وتوضيح الأجزاء العاتمة من الصورة للكشف المبكر عن نماذج السرطانات أو الكسور التي قد تكون مخفية أو عند حدوث خطأ في التصوير الشعاعي. تعتمد هذه العملية على جمع أو ضرب التدرج الرمادي مع قيمة ثابتة لزيادة إضاءة الصورة.
5. التعتيم: (Darkening) هي العملية المعاكسة لزيادة السطوع. في بعض الحالات، قد تكون الصورة عالية السطوع وتحتاج إلى درجة من التعتيم لتحسين توضيح الحواف والزوايا. تعتمد هذه العملية على طرح قيمة ثابتة من التدرج الرمادي في الصورة لتحريك التدرج باتجاه الصفر وتعتيم الصورة.
6. صورة التباين الجيبي العكسي: (Inverse Sinusoidal Contrast Image) هي عملية تهدف إلى دراسة وتحليل درجات التباين في الصورة. عادةً ما يتم استخدام مجموعة من الأحجام المستهدفة لاختبار حساسية التباين، والتي تختلف عن الحدة في أن حساسية التباين تقيس متغيرين: الحجم والتباين، بينما تقيس الحدة قيمة الحجم فقط. يعتمد اختبار حساسية التباين الحقيقي على أنماط جيبية تختلف في درجة السطوع عبر نمط الشبكة، ويتراوح سطوع الشبكة من 0.5% إلى 90%. تحدد حساسية التباين أدنى مستوى تباين يمكن للنظام اكتشافه لحجم معين من نمط الشبكة. تُعرف الشبكات ذات الأحجام المختلفة بالترددات المكانية [3] [22].

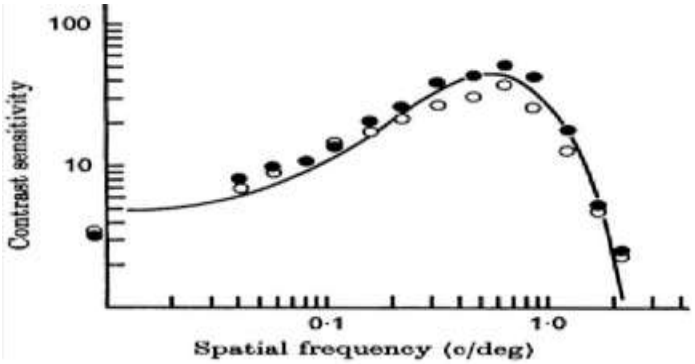
يُعرض منحنى حساسية التباين لتوضيح أدنى مستوى تباين يمكن اكتشافه لهدف محدد. يكون المحور الأفقي للمنحنى مخصصاً للتردد المكاني، بينما المحور العمودي مخصصاً لحساسية التباين. الترددات المكانية المنخفضة تمثل شبكات كثيفة، في حين أن الترددات المكانية العالية تمثل شبكات رقيقة. تعتبر حساسية التباين عكس مستوى التباين، فكلما زادت حساسية التباين، انخفض مستوى التباين الذي يمكن من خلاله اكتشاف الهدف. تُحسب حساسية التباين كعكس

عتبة التباين. على سبيل المثال، إذا تمكن النظام من اكتشاف حاجز شبكي عند تباين بنسبة 1% (المعروف بعتبة التباين)، فإن حساسية التباين عند هذا التردد المكاني تكون 100 أو عكس 1 % (1/.01)

$$contrast = \frac{Lmax - Lmin}{Lmax + Lmin}$$

حيث $Lmin$ تمثل المنطقة الأكثر تعتيماً و $Lmax$ المنطقة الأكثر سطوعاً.

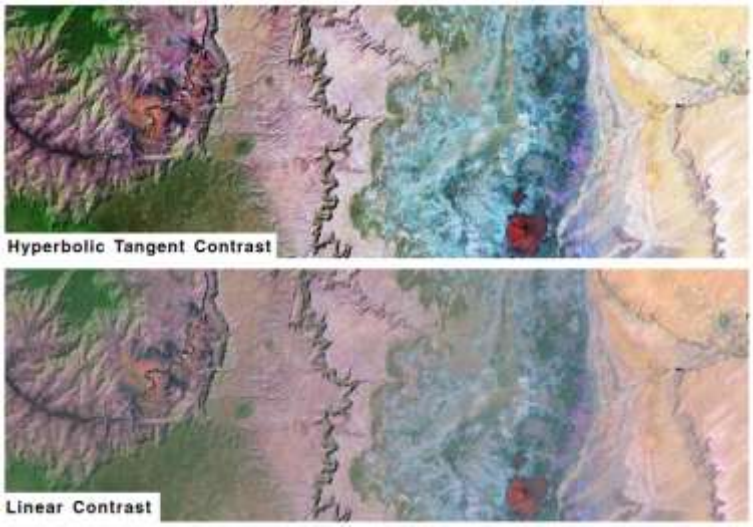
يوضح الشكل (3) التالي نموذجاً لهذا المخطط.




**الشكل (3) مخطط تعديل التباين الجيبي**

7. للتباين تأثير كبير على جودة الصورة في الإدراك البصري البشري. يمكن أن تؤثر البيئة ضعيفة الإضاءة بشكل كبير على التباين، مما يؤدي إلى صورة بجودة غير متوقعة. ولذلك، تُستخدم خوارزمية الظل الزائدي العكسي والعكسي التكيفي (AIHT)لتحسين جودة العرض وتباين المشهد. نظراً لأن الكاميرات الرقمية يجب أن تحافظ على الظل في نطاق متوسط من السطوع الذي قد يتضمن كائناً رئيسياً مثل الوجه، تُستخدم دالة غاما بشكل عام لهذا الغرض. ومع ذلك، تعاني هذه الدالة من ضعف في الأداء لأنها تقلل من تباين الإضاءة. وللتغلب على هذه المشكلة، تم تصميم خوارزميات تحسين التباين لتحسين التباين بحيث يتماشى مع الإدراك البصري البشري.

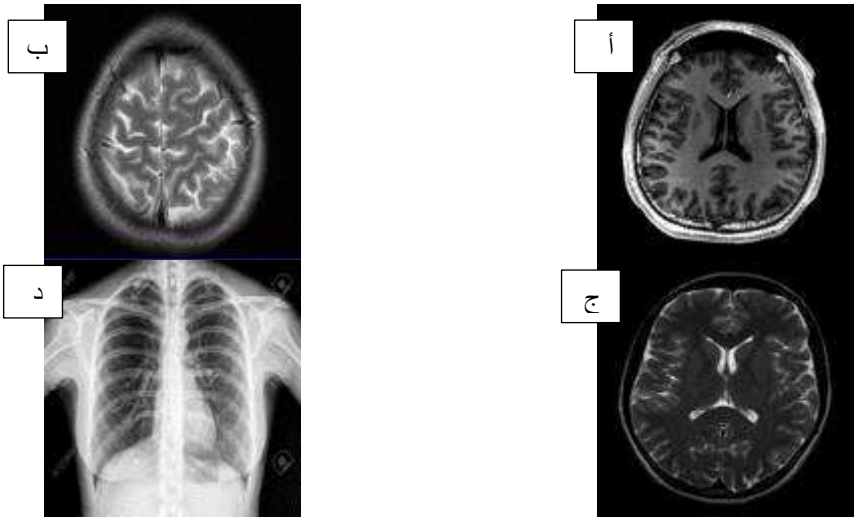
يحدد AIHT مستويات التباين للصورة الأصلية بالإضافة إلى مساحة المعلمات لأنواع التباين المختلفة، بحيث لا يتم الحفاظ على ميزات شكل الرسم البياني الأصلي فحسب، بل يتم أيضاً تحسين التباين بشكل فعال. تُظهر النتائج التجريبية أن الخوارزمية قادرة على تعزيز التباين الشامل للصورة الأصلية بشكل تكيفي مع إبراز تفاصيل الكائنات في وقت واحد. تستخدم طريقة التباين Hyperbolic Tangent منحني تحويل غير خطي لتحويل قيم البكسلات إلى قيم العرض. تؤدي زيادة قيمة المقياس لهذه الطريقة إلى تسوية الجزء المتوسط المدى من المنحني أثناء انحدار الأطراف المنخفضة والعالية.




**الشكل (4) الفارق بين تعديل التباين الظلي (أعلى) والتباين الخطي (أسفل)**

تمّ إجراء الاختبارات على أربع صور طبية ذات قيم تباين مختلفة ولأجزاء مختلفة من الجسم ومن ثم تطبيق عمليات معالجة الصورة المذكورة سابقاً عليها بشكل تسلسلي وتفرعي وبأحجام مختلفة، ثم مقارنة زمن المعالجة لكل منها.

يوضح الشكل (5) الصور التي تم اعتمادها في الدراسة.




**الشكل (5) الصور التي تم إجراء اختبارات معالجة الصورة عليها، الصورة الأولى (أ) صورة ذات تباين موسع للدماغ، الصورة الثانية (ب) صورة دماغية عالية السطوع، الصورة الثالثة (ج) صورة دماغية عاتمة، الصورة الرابعة (د) صورة للقفص الصدري.**

**السيناريو الأول: تجزئة السويات الرمادية**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.11769 | 0.69677 | 2.48808 | 10.86122 |
| Parallel Process | 0.000000002 | 0.000997 | 0.002990 | 0.007978 |

نسبة التسريع: 1183 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.087575 | 0.3111345 | 1.2107632 | 4.547035 |
| Parallel Process | 0.0009982 | 0.00099658 | 0.0039892 | 0.0119352 |

نسبة التسريع: 343 مرة

الصورة الثالثة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.070837 | 0.26550722 | 1.12994623 | 4.4981148 |
| Parallel Process | 0.0009973 | 0.0009973 | 0.00199484 | 0.00994563 |

نسبة التسريع: 428 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.082730 | 0.297266 | 1.102195 | 4.5145285 |
| Parallel Process | 0.000000002 | 0.000997 | 0.0000498 | 0.0148506 |

نسبة التسريع: 377 مرة

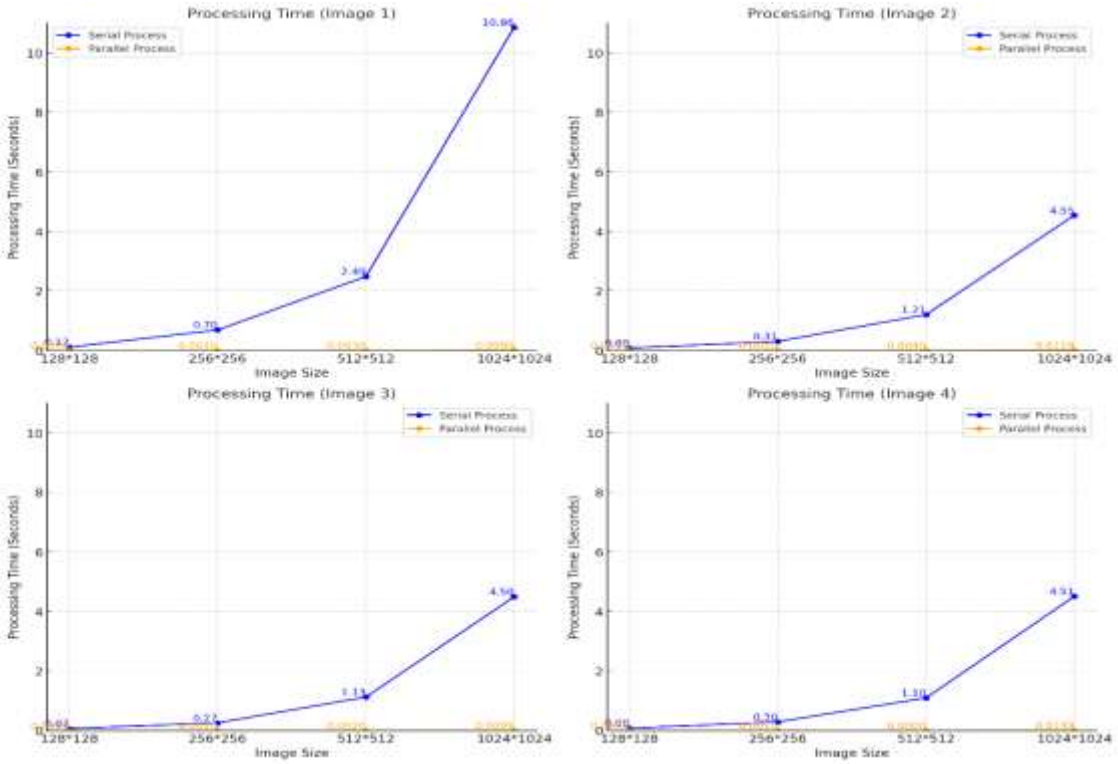
**طريقة حساب نسبة التسريع:**

**يتم حساب نسبة التسريع من خلال قسمة مجموع أزمنة التنفيذ التسلسلية على مجموع أزمنة التنفيذ التفرعية كما هو موضح في القانون الآتي:**

$$speedup = \frac{\sum serial\ times\ (in\ seconds)}{\sum parallel\ times\ (in\ seconds)}$$

**مثال يوضّح حساب نسبة التسريع للصورة الأولى:**

$$speedup = \frac{10.86122 + 2.48808 + 0.69677 + 0.11769}{0.007978 + 0.002990 + 0.000997 + 0.000000002} = 1183.77$$




**الشكل (6) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق تخفيض التباين**

**السيناريو الثاني: معكوس الصورة**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.1415910 | 0.5874557 | 2.3606603 | 10.49234 |
| Parallel Process | 0.0009980 | 0.0019936 | 0.0089762 | 0.026414 |

نسبة التسريع: 353 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.1286542 | 0.518614 | 2.329509 | 10.542106 |
| Parallel Process | 0.000000003 | 0.0009975 | 0.0039896 | 0.0159578 |

نسبة التسريع: 645 مرة

الصورة الثالثة:

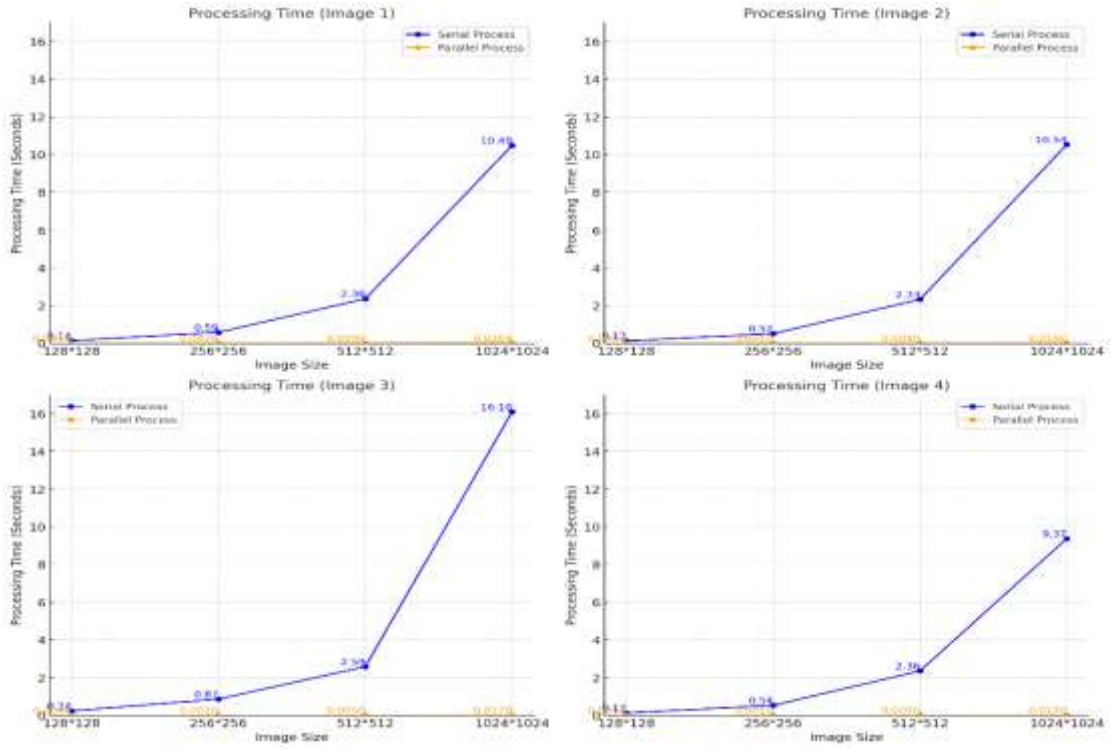
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.238362 | 0.867680 | 2.587086 | 16.095937 |
| Parallel Process | 0.000000004 | 0.001994 | 0.0049870 | 0.0169546 |

نسبة التسريع: 826 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.129655 | 0.535563 | 2.361364 | 9.374328 |
| Parallel Process | 0.000000004 | 0.000997 | 0.0049884 | 0.01699193 |

نسبة التسريع: 539 مرة




**الشكل (7) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق معكوس الصورة**

**السيناريو الثالث: تعتيب الصورة**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.0069816 | 0.0289154 | 0.12164950 | 0.961690 |
| Parallel Process | 0.000000001 | 0.000000003 | 0.0009996 | 0.0039916 |

نسبة التسريع: 186 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.0079491 | 0.03490376 | 0.12609601 | 0.4866988 |
| Parallel Process | 0.000000001 | 0.000000004 | 0.00196099 | 0.00398874 |

نسبة التسريع: 110 مرة

الصورة الثالثة:

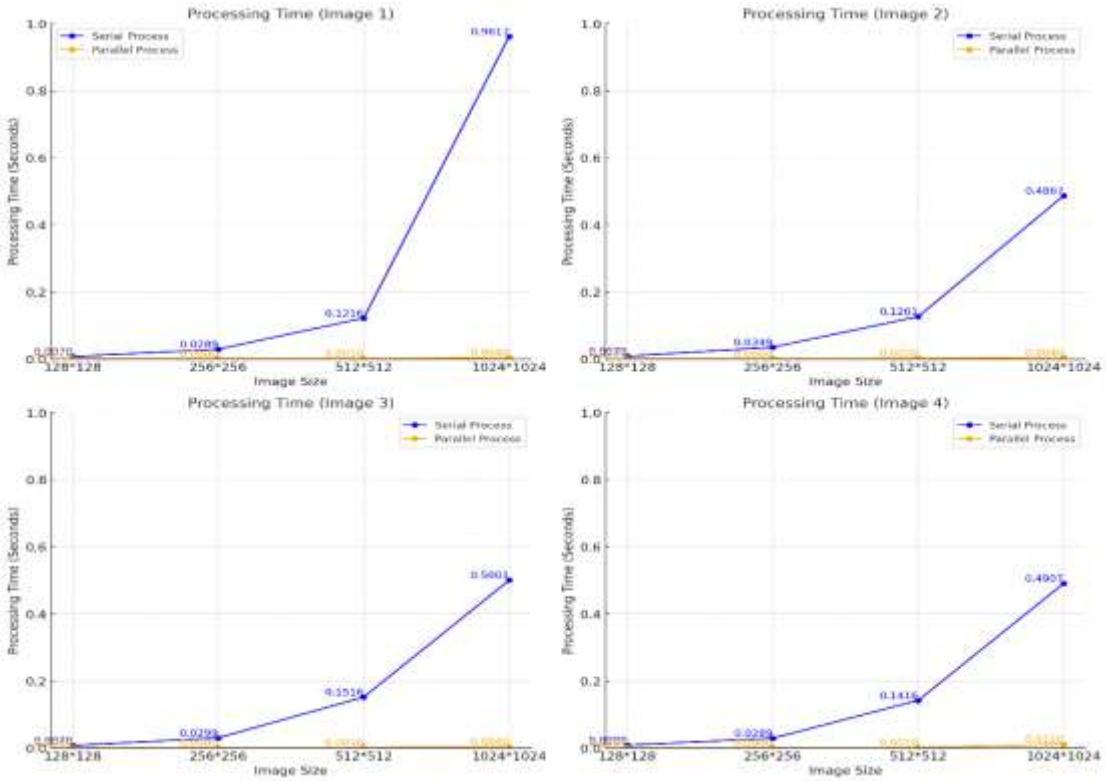
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.00698137 | 0.02992486 | 0.151557922 | 0.5000667 |
| Parallel Process | 0.000000001 | 0.000000006 | 0.00099706 | 0.00398921 |

نسبة التسريع: 138 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.0079774 | 0.02892589 | 0.14161968 | 0.49068808 |
| Parallel Process | 0.000000002 | 0.000000007 | 0.00099706 | 0.011004209 |

نسبة التسريع: 55 مرة




**الشكل (8) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق تعتيب الصورة**

**السيناريو الرابع: زيادة السطوع**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.1336102 | 0.59441018 | 2.4085633 | 10.2143716 |
| Parallel Process | 0.00199341 | 0.001994132 | 0.0079789 | 0.6585700 |

نسبة التسريع: 19 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.17054271 | 0.61339330 | 2.39020456 | 10.10477399 |
| Parallel Process | 0.00296044 | 0.002024888 | 0.00894522 | 0.668767452 |

نسبة التسريع: 19 مرة

الصورة الثالثة:

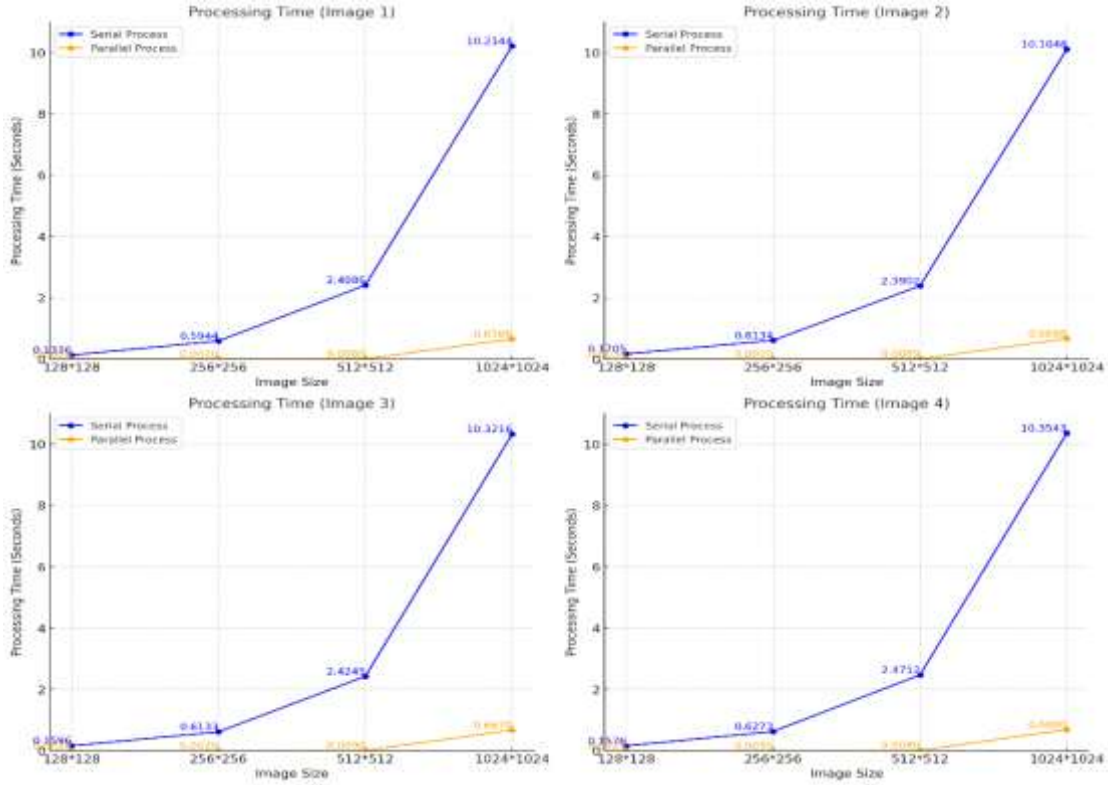
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.15957331 | 0.61332750 | 2.42451524 | 10.32155060 |
| Parallel Process | 0.00199270 | 0.00199365 | 0.009010314 | 0.666965 |

نسبة التسريع: 19 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.1575787 | 0.62732315 | 2.4712183 | 10.3542883 |
| Parallel Process | 0.00199460 | 0.00299191 | 0.00800991 | 0.688021421 |

نسبة التسريع: 19 مرة




**الشكل (9) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق زيادة السطوع**

**السيناريو الخامس: التعتيم**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.14960575 | 0.613359212 | 2.46397948 | 10.3426263 |
| Parallel Process | 0.0019629 | 0.002019405 | 0.00895333 | 0.638697385 |

نسبة التسريع: 20 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.14960145 | 0.610470533 | 2.4496445 | 10.5266060 |
| Parallel Process | 0.0030233 | 0.001993656 | 0.00894379 | 0.69324088 |

نسبة التسريع: 19 مرة

الصورة الثالثة:

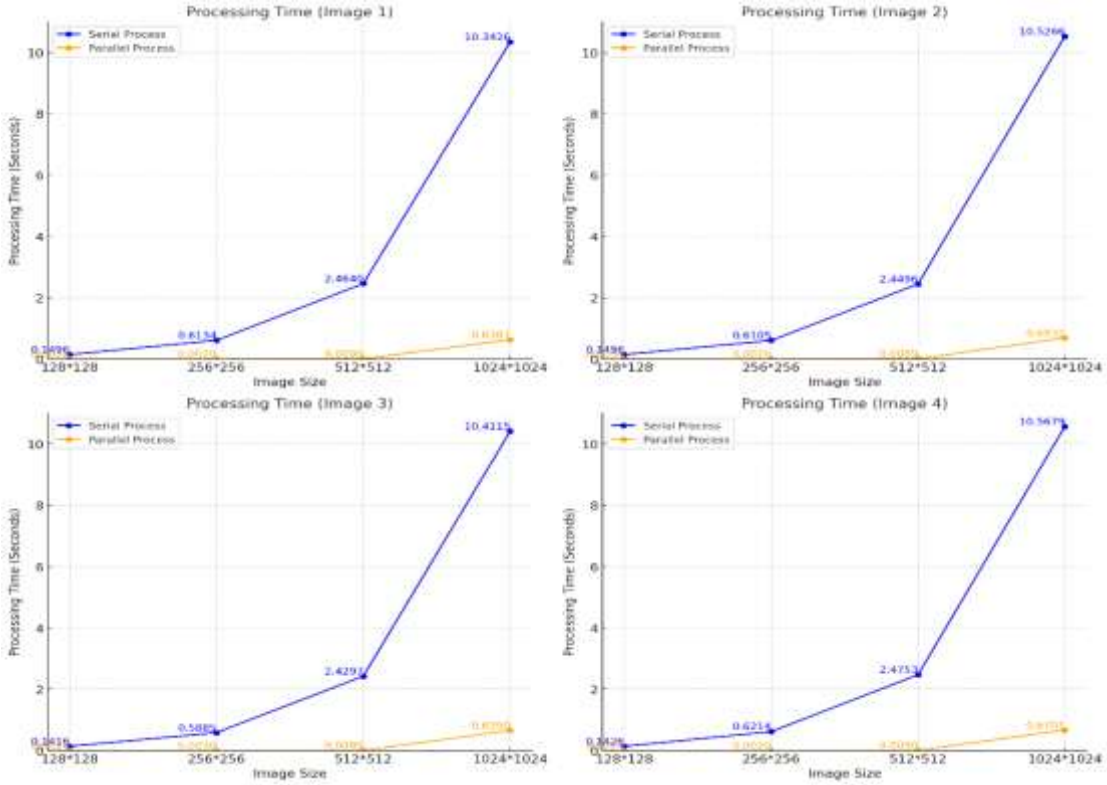
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.14162111 | 0.5884592 | 2.4296727 | 10.4114885 |
| Parallel Process | 0.0019946 | 0.00299215 | 0.0079779 | 0.658970117 |

نسبة التسريع: 20 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.14258742 | 0.6213700 | 2.47534966 | 10.56791377 |
| Parallel Process | 0.0020067 | 0.002017 | 0.0089738 | 0.67006206 |

نسبة التسريع: 20 مرة




**الشكل (10) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق تعتيم الصورة**

**السيناريو السادس: Inverse Sinusoidal contrast Image**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.4398245 | 1.22915101 | 4.98581409 | 23.4690752 |
| Parallel Process | 0.00203061 | 0.00199222 | 0.00794935 | 0.67792105 |

نسبة التسريع: 43 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.28125476 | 1.226683139 | 4.91589093 | 18.6332747 |
| Parallel Process | 0.001972675 | 0.002983093 | 0.007982730 | 0.676597833 |

نسبة التسريع: 36 مرة

الصورة الثالثة:

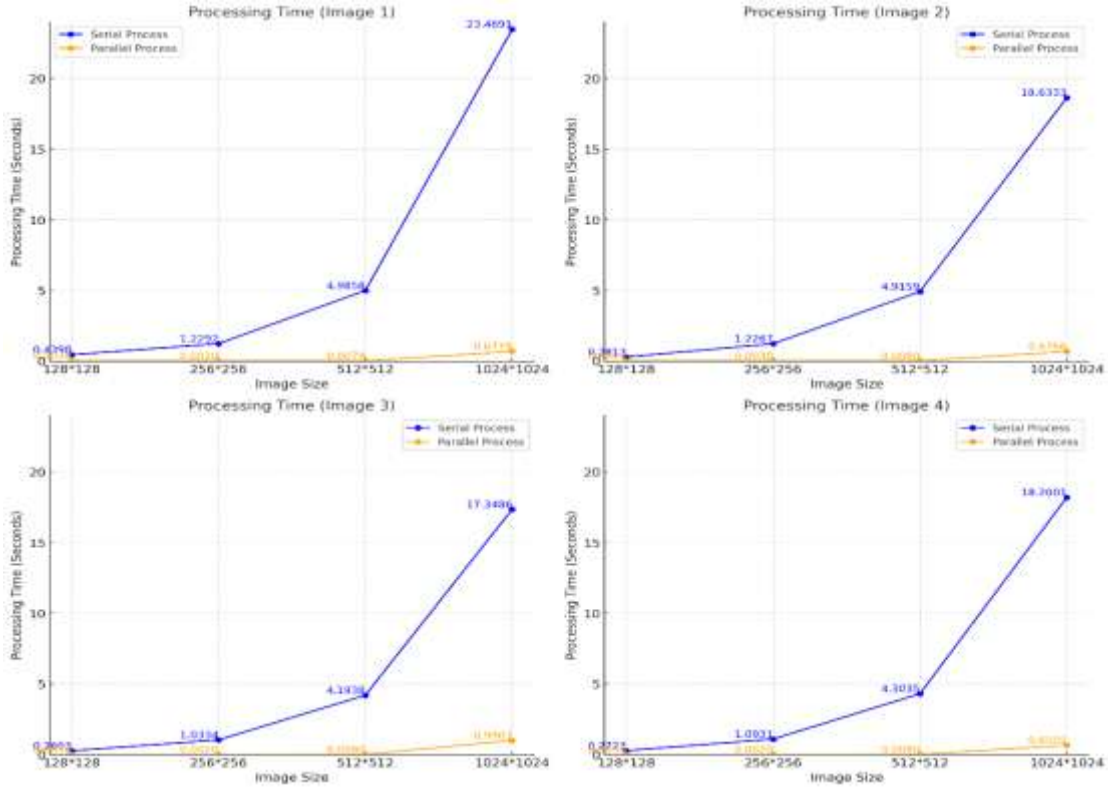
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.260304212 | 1.033404111 | 4.19378685 | 17.34860754 |
| Parallel Process | 0.003028154 | 0.001996755 | 0.007978200 | 0.9903154373 |

نسبة التسريع: 22 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.272309303 | 1.09308195 | 4.3034880 | 18.20012998 |
| Parallel Process | 0.0029602 | 0.002017498 | 0.007971286 | 0.650524616 |

نسبة التسريع: 35 مرة




**الشكل (11) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق ISC**

**السيناريو السابع: Hyperbolic Tangent Contrast Image**

الصورة الأولى:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.15274858 | 0.62532925 | 2.572376966 | 10.69406604 |
| Parallel Process | 0.001987934 | 0.00299358 | 0.00798344 | 1.20434093 |

نسبة التسريع: 11 مرة

الصورة الثانية:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.18350958 | 0.71013593 | 2.5325353 | 11.311711 |
| Parallel Process | 0.003021955 | 0.00199866 | 0.008975028 | 0.679842233 |

نسبة التسريع: 21 مرة

الصورة الثالثة:

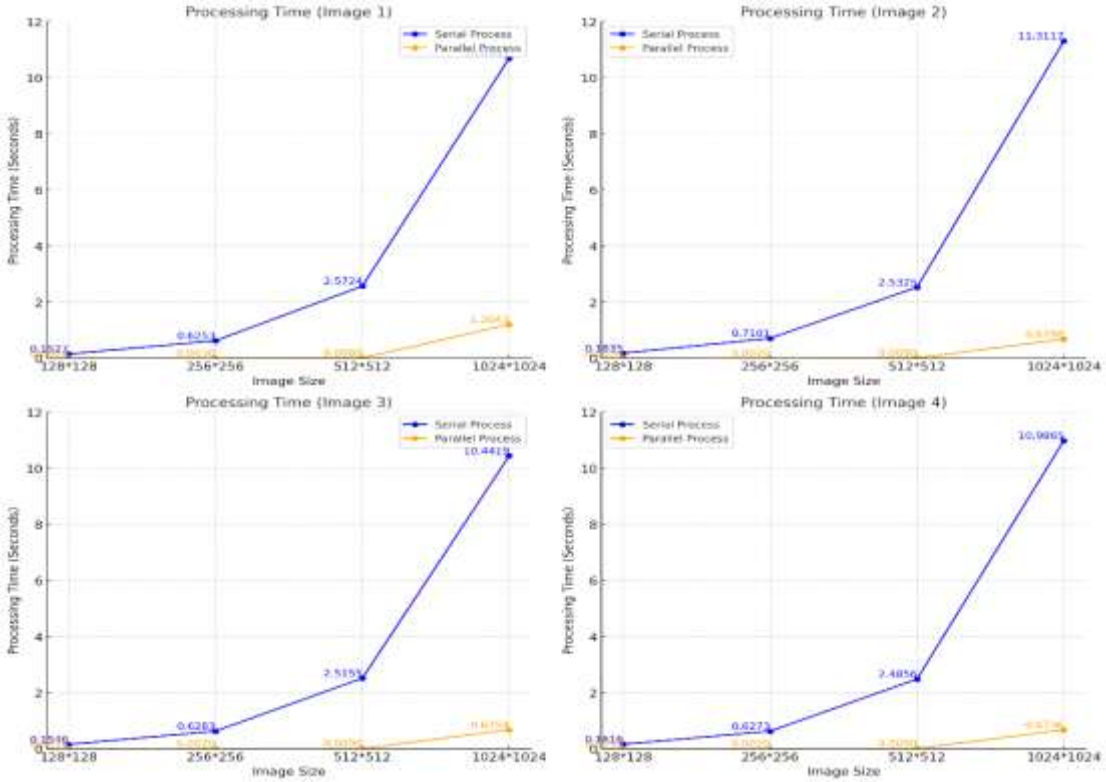
| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.15458869 | 0.62831974 | 2.51545977 | 10.441854 |
| Parallel Process | 0.002960681 | 0.001996755 | 0.00900816 | 0.67543244 |

نسبة التسريع: 19 مرة

الصورة الرابعة:

| حجم الصورة | 128*128 | 256*256 | 512*512 | 1024*1024 |
|---|---|---|---|---|
| Serial Process | 0.16156673 | 0.6273226 | 2.48556137 | 10.986495971 |
| Parallel Process | 0.00295543 | 0.001993894 | 0.00897622 | 0.67357468 |

نسبة التسريع: 20 مرة




**الشكل (12) مخططات مقارنة زمن المعالجة لأحجام مختلفة لتحقيق HTC**

## النتائج والمناقشة:

عند تحليل السيناريوهات المختلفة لمعالجة الصور الطبية باستخدام كل من وحدة المعالجة المركزية (CPU) ووحدة المعالجة الرسومية (GPU)، يتضح أن نسبة التسريع تعتمد بشكل رئيسي على تعقيد العملية وحجم الصورة. في السيناريو الأول (تجزئة السويات الرمادية)، حققت GPU أعلى نسبة تسريع (1183 مرة) بسبب الطبيعة المتوازية العالية للعملية، حيث يمكن تقسيم العمل على عدد كبير من الأنوية، خاصة عند معالجة الصور الكبيرة. أما السيناريو الثاني (معكوس الصورة)، فقد أظهر أيضًا كفاءة عالية مع نسب تسريع تصل إلى 826 مرة، حيث تستفيد العملية من الحسابات المستقلة لكل بكسل. في السيناريو الثالث (تعتيب الصورة)، كانت نسب التسريع أقل بكثير (55–186 مرة) نظرًا لأن العملية أبسط ولا تتطلب عمليات حسابية مكثفة، مما قلل من استفادة GPU بشكل كبير. أما السيناريو الرابع والخامس (زيادة السطوع والتعتيم)، فقد حققا أقل نسب تسريع (19–20 مرة) لأن العملية تتطلب فقط تعديل قيم البكسلات دون الحاجة إلى عمليات حسابية معقدة، مما جعل الأداء مقيداً بزمن نقل البيانات بين CPU و GPU أكثر من الحسابات نفسها. بالنسبة لـ السيناريو السادس (Inverse Sinusoidal Contrast) والسيناريو السابع (Hyperbolic Tangent Contrast)، فبينما تحقق GPU تحسينًا ملحوظًا (22–43 مرة)، إلا أن هذه العمليات قد تأثرت بعمليات نقل البيانات وإدارة الموارد، مما حد من الأداء المتوقع.

بشكل عام، أفضل تمثيل لتسريع GPU كان في السيناريو الأول (تجزئة السويات الرمادية)، حيث استفادت GPU بشكل كامل من بنيتها المتوازية، مما جعلها تتفوق بشكل كبير مقارنة بالـ CPU، لا سيما مع الصور ذات الدقة العالية.

**الاستنتاجات والتوصيات:**

بينت الدراسة والنتائج العملية الفعالية العالية لاستخدام وحدات CUDA في عمليات معالجة الصور الطبية، وقدرتها على إظهار النتائج بدقة وفي زمن يقارب الزمن الحقيقي مع تكلفة منخفضة.

عند تحليل نسب التسريع التي قدمتها وحدة GPU مقارنة بوحدة CPU في نماذج مختلفة من عمليات معالجة الصورة الأساسية التي تُستخدم بكثرة في المعالجة الطبية، تبين أن نسبة التسريع تتجاوز 500 مرة، مما يسمح للطبيب بالحصول على نتائج دقيقة وسريعة تساعد في فهم حالة المريض.

تُعد وحدات المعالجة GPU ثورة في مجال الطب من حيث الدقة في عمليات المعالجة، بالإضافة إلى السرعة العالية، حتى في مجال استخدام الشبكات العصبية التي يمكن تنفيذها أيضاً على وحدات GPU لتحقيق التنبؤ بحالة المريض في زمن يقارب الزمن الحقيقي مقارنة بوحدات المعالجة المركزية.

يمكن إجراء تجارب إضافية باستخدام وحدات GPU بقدرات أعلى (مثل وحدات تحتوي على آلاف الأنوية) لمقارنة الأداء مع الأنظمة المستخدمة في المستشفيات والمراكز البحثية المتقدمة.

يُقترح دمج تقنيات التعلم العميق والشبكات العصبية باستخدام وحدات GPU لتحسين دقة التشخيص والتنبؤ بالحالات المرضية، مع التركيز على تطبيقات مثل اكتشاف الأورام، وتتبع نموها، وتحليل النتائج بشكل آلي.

يُنصح بإجراء تجارب باستخدام أحدث إصدارات تقنيات CUDA التي توفر تحسينات في الأداء والبرمجة التفرعية، للاستفادة من الميزات الجديدة مثل تحسين إدارة الذاكرة ودعم العمليات الحسابية المعقدة.

## References:

1. Yi X, Qiao Y. GPU-Based Parallel Computing Methods for Medical Photoacoustic Image Reconstruction. 2024 Apr 16.
2. Kirimtat A, Krejcar O. GPU-Based Parallel Processing Techniques for Enhanced Brain Magnetic Resonance Imaging Analysis: A Review of Recent Advances. Sensors. 2024 Feb 29;24(5):1591.
3. Singh N. Accelerate Image Processing Tasks With Nvidia GPUs. 2024 Jan 17.
4. Santosh KC, Goyal A, Aouada D, Makkar A, Chiang Y-Y, Singh SK. Recent Trends in Image Processing and Pattern Recognition: 5th International Conference, RTIP2R 2022, Kingsville, TX, USA, December 1-2, 2022, Revised Selected Papers. 2023 Jan 11.
5. Meyer J. Evaluation of Modern GPGPU Technologies for Image Processing. Proceedings of the International Workshop on OpenCL (IWOCL '20). 2020 Apr 27.
6. Packt Publishing. Hands-On GPU-Accelerated Computer Vision with OpenCV and CUDA. 2018 Sep 27.
7. Ait Ali N, Cherradi B, El Abbassi A, Bouattane O, Youssfi M. GPU Fuzzy C-Means Algorithm Implementations: Performance Analysis on Medical Image Segmentation. Springer. 2018.
8. Kalaiselvi T, Sriramakrishnan P, Somasundaram K. Performance of Medical Image Processing Algorithms Implemented in CUDA Running on GPU-Based Machine. 2018.
9. Amalia AE, Airlangga G, Thohari ANA. Breast Cancer Image Segmentation Using K-Means Clustering Based on GPU CUDA Parallel Computing. JURNAL INFOTEL Informatics - Telecommunication - Electronics. 2018.

10. Oza S, Joshi KR. CUDA-Based Fast Bilateral Filter for Medical Imaging. Proceedings of the 5th International Conference on Signal Processing and Integrated Networks (SPIN). 2018.
11. Li CY, Chang HH. CUDA-Based Acceleration of Collateral Filtering in Brain MR Images. Proceedings of the 8th International Conference on Graphic and Image Processing (ICGIP). 2017;10225:1022528.
12. Fluck O, Vetter C, Wein W, et al. A Survey of Medical Image Registration on Graphics Hardware. Comput Methods Programs Biomed. 2011;104:e45-e57.
13. Owens JD, Houston M, Luebke D, et al. GPU Computing. Proc IEEE. 2008;96:879-99.
14. Yang ZY, Zhu YT, Pu Y. Parallel Image Processing Based on CUDA. Proceedings of CSSE. 2008;198-201.
15. Collins R, Li CH, Carloni LP, et al. An Experimental Analysis of General Purpose Computing with Commodity Data-Parallel Multicore Processors. Technical Report RC25070, IBM T. J. Watson Research Center. 2010.
16. Schellmann M, Gorlatch S, Meilander D, et al. Parallel Medical Image Reconstruction: From Graphics Processing Units (GPU) to Grids. J Supercomput. 2011;57:151-60.
17. Huang TY, Tang YW, Ju SY. Accelerating Image Registration of MRI by GPU-Based Parallel Computation. Magn Reson Imaging. 2011;29:712-6.
18. Bui P, Brockman J. Performance Analysis of Accelerated Image Registration Using GPGPU. Proceedings of GPGPU. 2009;38-45.
19. Archirapatkave V, Sumilo H, See SCW, et al. GPGPU Acceleration Algorithm for Medical Image Reconstruction. Proceedings of ISPA. 2011;41-6.
20. Schenke S, Wuensche B, Denzler J. GPU-Based Volume Segmentation. Proceedings of IVCNZ. 2005;171-6.
21. Chen HLJ, Samavati FF, Sousa MC, et al. Sketch-Based Volumetric Seeded Region Growing. Proceedings of Eurographics. 2006;123-9.
22. Bojsen-Hansen M. Active Contours Without Edges on the GPU. Project Paper for the Course in Parallel Computing for Medical Imaging and Simulation. 2010;1-8.
23. Roberts M, Packer J, Sousa MC, et al. A Work-Efficient GPU Algorithm for Level Set Segmentation. Proceedings of HPG. 2010;123-32.
24. Chen HLJ, Samavati FF, Sousa MC. GPU-Based Point Radiation for Interactive Volume Sculpting and Segmentation. Visual Comput. 2008;24:689-98.
25. Feltell D, Bai L. Level Set Brain Segmentation with Agent Clustering for Initialization. Proceedings of BIOSIGNALS. 2008;1-8