وزارة التعليم العالي والبحث العلمي

جامعه بابل كليه العلوم للبنات

قسم علوم الحاسوب

ضغط البيانات

بحث مقدم الى مجلس كليه العلوم للبنات – قسم علوم الحاسوب

وهو جزء من متطلبات نيل شهادة البكالوريوس في علوم الحاسوب

اعداد الطالبة

رسل حيدر عزيز

بأشراف الدكتور

سيف العلاق

١٤٤٥ه

٢٠٢٤م

بِسْمِ اللَّـهِ الرَّحْمَـٰنِ الرَّحِيمِ

{يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ

دَرَجَاتٍ} المجادلة : الاية١١

صدق الله العلي العظيم

الاهداء

إلى ربنا الذي علمنا كل حرف عرفناه..

إلى مدينة العلم وبابها الذي يؤتى النبي (ص) والوصي

إلى سفن النجاة وأعلام الهدى الائمة الأطهار(ع)..

إلى من جرع الكأس فارغاً ليسقينا قطرة حب إلى من كلت أنامله

ليقدم لنا لحظة سعادة إلى من حصد الشواك عن دربنا ليمهد لنا

طريق العلم إلى والدي الغالي

إلى من كانت سندي في مسيرتي الدراسية والذي وقفت بجانبي في

أزمات حياتي الوالدة الحبيبة ...

الى نجوم سمائي المتألقة سندي في الحياة أخوتي ...

الى سند المستقبل أساتذتي وزملائي...

الى كل قلب خفق لي حبًا وخوفًا علي...

اسال الله ان يفتح لي ولأصدقائي ابواب الخير والتوفيق

أجمل لحظة هي أن يتحقق ما صبرت وتعبت لأجله

الشكر والتقدير

الحمد لله رب العالمين والصلاة والسلام على معلم البشرية ،هادي الإنسانية وعلى اله الطيبين الطاهرين واصحابه المنتجين.

نتوجه بالشكر الجزيل:

الى الكهف الحصين وغياث المضطر المستكين ملاذ المؤمنين صاحب العصر والزمان الإمام المهدي

الى من أستطيع أن أقابله بالشكر والإحسان فكل شكر اليه ، هو بالحقيقة منه واليه و الى كل من ساهم في اخراج هذا البحث،

الى حيز التنفيذ الى كل من كان سبباً في تعليمنا وتوجيهينا ومساعدتنا

الى الدكتور الفاضل :- د.سيف العلاق

ت	العناوين	الصفحة
١	**الاية**	١
٢	**الاهداء**	٢
٣	**الشكر والتقدير**	٣
٤	**المقدمة**	٥
٥	**الفصل الأول:** الجانب النظري	٦ـ ١٢
٦	**الفصل الثاني:** الجانب العملي	١٣ـ ١٥
٧	المصادر	١٦

المقدمة

يعتبر ضغط البيانات أمرًا مفيدًا نظرًا لأنه يساعد على خفض استخدام الموارد، مثل وسط التخزين أو سعة الإرسال أو حيز التردد. ونظرًا لأن البيانات المضغوطة لابد من فك ضغطها لكي تصبح صالحة للاستخدام، فإن عمليات المعالجة الإضافية هذه تضع قيودًا حسابية أو تكلفة من نوع آخر لعملية فك الضغط، وبالتالي فإن هذه العملية ليست بلا ثمن!. ويتطلب فك ضغط البيانات الموازنة بين الوقت/حيز التخزين من ناحية وبين التعقيد من ناحية أخرى. فعلى سبيل المثال، عند استخدام أحد أساليب ضغط الفيديو، يتطلب الأمر إمّا أجهزة مكلّفة لكي تتمكن من فك ضغط الفيديو أثناء المشاهدة (وهو ما يزيد التعقيد)، أو أن يتم فك الضغط بالكامل قبل المشاهدة وهو ما يكلِّف وقتًا (في أنظمة الإرسال/الاستقبال) ويتطلب أيضًا حيز تخزين كبير يستوعب الفيديو الذي تم فكّه. ويعتمد تصميم أنظمة ضغط البيانات على الموازنة بين عدة عوامل مختلفة، والتي تشمل درجة الضغط، وكمية التشويه (Distortion) التي يمكن احتمالها (وذلك عند استخدام أنظمة الضغط المنقوص(، وكمية الموارد الحسابية المطلوبة لفك ضغط البيانات.[٥] وتجدر الإشارة إلى أن نظرية المعلومات هي العلم الذي يبحث في كلا نوعي ضغط البيانات، كما سيأتي تفصيله لاحقًا، حيث يهدف الضغط غير المنقوص إلى تمثيل البيانات بعدد من النبضات الثنائية يكافئ بالضبط الإنتروبية بينما يؤدي الضغط المنقوص إلى تمثيل البيانات بأقل من الإنتروبية، وبالتالي يحدث فقد في المعلومات (تشويه) وتبحث النظرية حينئذٍ في تحقيق أقصى استفادة مناظرة لهذا التشويه.

الفصل الاول: الجانب النظري

مفهوم ضغط البيانات

في مجال معالجة الإشارات الرقمية، يشير مصطلح ضغط البيانات (Compression Data)

أو الترميز المصدري(Coding Source)، أو خفض معّدل البيانات إلى ترميز المعلومات باستخدام قدر أقل من النبضات الثنائية من التمثيل الأصلي. يمكن أن يكون ضغط البيانات إمّا ضغط منقوص أو ضغط البيانات غير المضيع بالإنجليزية (Loss) اوLossless (Compression)على الترتيب. الضغط غير المنقوص يخفض من عدد النبضات الثنائية البت اللازمة للتعبير عن البيانات عن طريق التعرف على التكرارية الإحصائية .وال تفقد أي معلومات نتيجة لضغط البيانات. ومن أمثلة هذا النوع البرامج التي تستخدم لضغط ملفات الحاسوب مثل برامج ليتم تخزينها في صيغة (*.rar)أو في صيغة (*.zip).ويتم استعادة ملفات الحاسب أيًا كان نوعها بالكامل بعد فك الضغط. أمّا في حالة الضغط المنقوص فيتم تقليل النبضات الثنائية من خلل التعرف على المعلومات غير الهامة أو غير الأزمة وإزالتها. ومن أمثلة هذا النوع ما يحدث عند تخزين الصور ومقاطع الفيديو بصيغ تقلل الحجم، حيث يتم ضغط بقدر غير محسوس، مثال: تحويلها من صيغة حجم الملفات عن طريق تقليل جودة الصورة **مثالً**

(*.bmp)الكاملة إلى صيغة (*.jpg) المضغوطة. وكذلك الحال عند تحويل صيغة (*.avi)

الكاملة إلى صيغ أخرى مثل)*٤.mp)المضغوطة. وال يمكن استرجاع »المعلومات« المفقودة عند الضغط بهذه الأساليب

وتسمى عملية خفض حجم ملف البيانات باسم ((ضغط البيانات)). أمّا في سياق نقل البيانات وفي أنظمة الاتصالات، فيطلق عليها »ترميز المصدر حيث يتم الترميز عند مصدر البيانات قبل تخزينها أو إرسالها وهذا على العكس من ترميز القناة. والذي يقصد منه تصحيح الأخطاء التي تسببها القناة أثناء الإرسال. تسببها القناة أثناء الإرسال.

ضغط البيانات أو مصدر الترميز بالإنجليزية: (compression Data)في علوم الحاسوب والمعلومات النظرية، هو عملية تشفير المعلومات حيث تأخذ حيزا قليلا من المساحة. ويوجد العديد من البرامج التي تقوم بضغط البيانات من أشهرها ويب

اختلاف البيانات

يمكن اعتبار ضغط البيانات كحالة خاصة لـ اختلاف البيانات. يتكون اختلاف البيانات من إنتاج "اختلاف" مع إعطاء "مصدر" و"هدف" ، مع تسليم إعادة إنتاج "الهدف" مع إعطاء "مصدر" و"اختلاف". نظرًا لعدم وجود مصدر أو هدف منفصل في ضغط البيانات ، يمكن للمرء حتى يعتبر ضغط البيانات اختلافًا عن البيانات مع بيانات المصدر الفارغة ، ويتوافق الملف المضغوط مع اختلاف عن لا شيء. هذا هونفس اعتبار المطلقة entropy وهو مقياس

لوگاريتمي لمعدل نقل المعلومات في رسالة أولغة معينة.(اللقاءة لضغط البيانات) كحالة خاصة لـ الإنتروپيا النسبية (اللقاءة لاختلاف البيانات) بدون بيانات أولية.

ما المقصود من ضغط البيانات؟

ضغط البيانات أو ما تسمى بـCompaction، هي عملية هدفها التقليل من كمية البيانات التي

نحتاج إلى تخزينها في إحدى وسائط التخزين، أو التي تحتاج إلى نقل عبر الشبكة، وذلك من أجل تسريع عملية الإرسال ويتم ذلك عادة باستخدام تقنية الترميز.

أنواع ضغط البيانات :

١- الضغط بدون فقدان البيانات: يستخدم هذا النوع من الضغط للحفاظ على جودة البيانات الأصلية دون أي فقدان في الجودة، ومن أمثلة هذه التقنيات الضغط بالتعريفات والضغط بالترميز.
٢-الضغط بفقدان البيانات: يستخدم هذا النوع من الضغط للتخلص من المعلومات غير المهمة والتي ال تؤثر على جودة البيانات النهائية، ومن أمثلة هذه التقنيات الضغط بالصور والضغط بالصوت.
٣ -الضغط المتعدد المستويات: يستخدم هذا النوع من الضغط لتحقيق أعلى معدلات الضغط الممكنة من خلال استخدام تقنيات الضغط بدون فقدان البيانات والضغط بفقدان البيانات معًا، ومن أمثلة هذه التقنيات الضغط المتعدد المستويات للصور والفيديو.

خوارزمية ضغط الملفات

نتحدث عن خوارزمية ضغط الملفات
خوارزمية هوفمان توفر من ٢٠% إلى ٩٠% من حجم الملف!! وذلك اعتمادا على نوع الملف تعمل بكفاءة عالية وتوفر كثيراً نفسه، حيث أن خوارزمية هوفمان من حجم الذاكرة في حالة الملفات النصية ،ولكن لها تأثيرها ليس بكبير على ملفات الصور والفيديو مقارنة بالملفات النصية

ليكن لدينا النص

Abcabcabca

هذا النص سيخزن في ذاكرة حجمها ١٠*٨ =٨٠ بت، لماذا؟ ألن كل حرف يأخذ ٨ بتات، و أحرف، و يُعد هذا إسرافاً النص يتكون من ١٠ في استخدام الموارد، بالمكان أن يُخزن كل حرف

في ٢ بت فقط كالاتي :

الحرف البت

a ٠١
b ١٠
c ١١

أي أن الملف قابل للتخزين في ١٠*٢=٢٠ بت وهذا فرق كبير حيث قمنا بتخزين ملف كان يأخذ ٨٠ بت في ٢٠ بت فقط.
مكررة ٤ مرات، aعن طريق بت واحد فقط وهو الـ ٠، وبما أن الـ aكذلك بالمكان تمثيل الـ عن طريق بت واحد فقط نكون قد وفرنا ٤ بت أخرى ، أي أنه يمكننا حفظ aبذلك عند تمثيل الـ الملف في ذاكرة حجمها ١٦ بت فقط بدال من ٢٠ بت.

السؤال الذي يطرح نفسه هل يشكل فرق أن مثلنا الحرف

هو ٣ فقط، b؟ الجواب: نعم يُشكل فرقاً ،وذلك ألن تكرار حرف الـ a بدً ال بالبت ٠ عن الـ b بالـ ٠. aأي أننا سنوفر ٣ بت فقط وهي أقل من التي نوفرها في حالة مثلنا حرف الـ [box type=”info” align=”aligncenter” class=”” width=””] إلى نخلص هذا من بأقل عدد من البتات [boxنكون قد وفرنا ذاكرة أكبر.]/ (Bitsأنه كلما مثلنا العدد الأكثر تكرارا

(Huffman Treeً) :

بدايةً وفمان قبل القيام بضغط ملف أو فك الضغط عنه يجب بناء شجرة

كيف ننشئ شجرة وفمان؟!!
Abeecbedcbaeddcbeeabeaليكن لدينا النص أوً ال إحسب تكرار كل حرف
e=7،d=3،c=3،b=5،A=4

بعدد الحرف التي لديك، أي كون
كّون أوراقا ٥ أوراق (عقد)و ضع في كل ورقة تكرار الحرفً الذي تمثله.

اختر عقدتان لهما أصغر تكرارين. اجعلها ابنتان لعقده جديدة.

هذه العقدة الجديدة ضع فيها مجموع التكرارين (كراري) العقدتين صاحبتا أقل تكرارين(.

طبق هذه العملية حتى تحصل على عقدة واحده وهي الجذر. لموضعها، إذا كانت يمين عقدة أخرى تأخذ الرقم رق ١ وإذا كانت شمال عقدةّم كُل عقده وفقاً أخرى تأخذ الرقم ٠

بهذه الست خطوات تكون قد أنشئت شجرة هوفمان)و بالإمكان تطبيق عمليتي ضغط الملفات وفك الضغط عليها

ضغط البيانات:

بعد إنشاء شجرة هوفمان وترقيم كل عقدة، في عملية الضغط نمثل كل حرف بأرقام العقد التي

نقابلها في طريقنا من العقدة التي تحوي تكرار الحرف المعني حتى الجزر. علي سبيل المثال في

الشكل اعاله رقم(a)نمثل الحرف a بالبتات .٠٠٠

أقل من البتات،

والحرف e بالبتات ٠٠١ ونلاحظ أن الشجرة تعطي الحرف الأكثر تكراراً عدداً وهكذا قمنا بالضغط عن طريق خوارزمية وفمان.

فك الضغط :

لفك الضغط، في بادئ الآمر يكون لدينا سلسلة بتات ناتجة من الضغط. أول خطوة قف عند الجذر. أنظر إلى سلسلة البتات الناتجة من الضغط و انظر إلى أول بت إذا كان ٠ اذهب إلى العقدة يسار الجذر و إذا كانت ١ اذهب للعقدة يمين الجذر اذهب يسار العقدة التي تقف
أو يسارا أنظر إلى البت الثاني إذا كان صفراً في كل تحرك يميناً
عليها وإذا كان ١ اذهب يمين العقد التي تقف عليها. أستمر هكذا حتى تصل إلى ورقة و اكتب الحرف الذي تمثله. عد و كرر نفس العملية حتى تصل إلى آخر بت.
وهكذا تكون قد وصلت إلى (النص المطلوب)أي فككت الضغط

مثال

يشرح هذا المثال كيفية عمل الخوارزمية بشكل عملي، حيث يظهر ويراقب الدخل والخرج والقاموس في كل مرحلة من مراحل العمل الترميز وفك الترميز. هذا المثال معمول خصيصاً ليظهر نسبة ضغط مقبولة على نص قصير. في النصوص الحقيقية ال يظهر التكرار بهذا الشكل وال يمكن أن تصل نسبة الضغط لهذا المستوى على نصوص بهذا الطول.
النص المراد تشفيره مأخوذ من صورة تحوي الألوان الأساسية الثالث (ألحمر، الأخضر والأزرق) وهو:

ح ح ح خ خ خ ح ح خ خ ز ز ز ز ز ز ز ز ز ز ز ز ز ز ز#

يشير محرف # إلى نهاية النص. بما أن هذه الرسالة مأخوذة من صورة تحوي الألوان الثالثة فإن الرموز الممكنة هي ح وخ وز وبالتالي فإن القاموس الأولي سيحوي الرموز الثالثة وسيحوي على رمز نهاية النص. وبالتالي سيكون القاموس المبدئي على الشكل:

٠ = ٠٠ = #
ح = ٠١ = ١
خ = ١٠ = ٢
ز = ١١ = ٣

الترميز

يقوم المرمز بمخبئة المحارف من سلسلة الدخل ضمن السلسلة ωحتى تصبح السلسلة + ω

المحرف التالي غير موجودة ضمن القاموس. يتم إخراج القيمة الموافقة للسلسلة ωضمن القاموس من المحرف ويتم إضافة السلسلة + ωالمحرف التالي إلى القاموس. ومن ثم بدء المخبئة ابتدا التالي.

السلسلة المحرف الخرج القيم المضافة

الحالية التالي القيمة البتات للقاموس

---- ---- ---- ----

»ال شيء« ح

ح ح ١ ٠٠٠١ :٤ ح ح □- القيمة ٤ هي أول قيمة فارغة ضمن القاموس

القيم من ٠ وحتى ٣

بما أنه يحوي مبدئياً

ح ح

ح ح خ ٤ ٠١٠٠ :٥ ح ح خ

خ خ ٢ ٠٠١٠ :٦ خ خ

خ خ

خ خ ح ٦ ٠١١٠ :٧ خ خ ح

ح ح

ح ح خ

ح ح خ خ ٥ ٠١٠١ :٨ ح ح خ خ

خ ز ٢ ٠٠١٠ :٩ خ ز

ز ز ٣ ٠٠١١ :١٠ ز ز

ز ز

ز ز ز ١٠ ١٠١٠ :١١ ز ز ز

ز ز

ز ز ز

ز ز ز ز ١١ ١٠١١ :١٢ ز ز ز ز

ز ز

ز ز ز

ز ز ز ز

ز ز ز ز ز ١٢ ١١٠٠ :١٣ ز ز ز ز ز

ز ز

ز ز ز

ز ز ز ز

ز ز ز ز ز

ز ز ز ز ز # ١٣ ١١٠١

طول البيانات قبل الترميز: ٢٥ محرف × ٢ بت للمحرف = ٥٠ بت

طول البيانات بعد الترميز: ١١ قيمة × ٤ بت للقيمة = ٤٤ بت

باستخدام خوارزمية إل زد دبليو تم التخلص من ٦ بتات، أي أن النص قد تم ضغطه بنسبة ١٢%،

طبعا ٤ رموز فقط بينما في حال تمً هذه النسبة ضئيلة هنا ألن اآل)استخدام أبجدية أكبر كجدول أسكي مثالً فإن النسبة سترتفع ألن عدد البتات اللازمة للمحرف قبل

الترميز سيكون أكبر من عدد البتات اللازمة للقيمة بعد الترميز.
تشفير Huffman التكيفي

نوع مختلف يسمى تشفير Huffman التكيفي يتضمن حساب الاحتمالات ديناميكيًا بناءً على الترددات الفعلية الأخيرة في تسلسل رموز المصدر ، وتغيير هيكل شجرة التشفير لمطابقة تقديرات الاحتمالات المحدثة. نادرًا ما يتم استخدامه في الممارسة العملية ، نظرًا ألن تكلفة تحديث الشجرة تجعلها أبطأ من الترميز الحسابي التكيفي ، وهو أكثر مرونة ولديه ضغط أفضل.

خوارزمية قالب Huffman

في أغلب الأحيان ، تمثل الأوزان المستخدمة في تطبيقات ترميز هوفمان الاحتمالات الرقمية ، لكن الخوارزمية المذكورة أعاله ال تتطلب ذلك ؛ يتطلب فقط أن تشكل الأوزان مرتبة تمامًا أحادية تبادلية ، مما يعني طريقة لترتيب الأوزان وإضافتها. تمكّن خوارزمية قالب هوفمان الشخص من استخدام أي نوع من الأوزان التكاليف والترددات وأزواج الأوزان والأوزان غير الرقمية وإحدى(طرق الجمع العديدة)وليس الإضافة فقط يمكن لمثل هذه الخوارزميات أن تحل مشاكل التصغير الأخرى ، مثل تصغير ، تم تطبيق مشكلة ألول مرة على تصميم الدوائر.

ترميز Huffman

محدود الطول هو متغير حيث ال يزال الهدف هو تحقيق الحد الأدنى لطول المسار الموزون ، ولكن هناك قيد إضافي يجب أن يكون طول كل كلمة مشفرة أقل من ثابت معين. تحل خوارزمية دمج الحزم هذه المشكلة بطريقة بسيطة جشع تشبه إلى حد بعيد تلك المستخدمة بواسطة خوارزمية Huffman. تعقيده الزمني هو ، حيث هو أقصى طول لكلمة الرمز . ال توجد خوارزمية معروفة لحل هذه المشكلة في أو الوقت ، بخالف مشاكل هوفمان التقليدية التي تم فرزها مسبقًا والتي لم يتم فرزها ، على التوالي.
تشفير huffmanبتكاليف أحرف غير متساوية
في مشكلة ترميز Huffman القياسية ، يُفترض أن كل رمز في المجموعة التي يتم إنشاء كلمات الكود منها له تكلفة متساوية للإرسال: رمز الكلمة التي يكون طولها N عدًدا سيكون لها دائمًا تكلفة N ، بغض النظر عن عدد هذه الأرقام التي تكون s٠ ، وكم عدد s١ ، وما إلى ذلك. عند العمل وفقًا لهذا الافتراض ، يتم تقليل التكلفة الإجمالية للرسالة وتقليل إجمالي عدد الأرقام هي نفس الشيء.
ترميز huffman بتكاليف غير متساوية للحروف هو التعميم بدون هذا الافتراض: قد يكون أحرف أبجدية التشفير أطوال غير موحدة ، بسبب خصائص وسيط الإرسال. مثال على ذلك هو أبجدية الترميز الخاصة بـ شفرة مورس ، حيث يستغرق إرسال "شرطة" وقتًا أطول من "نقطة" وبالتالي فإن تكلفة شرطة وقت الإرسال تكون أعلى. ال يزال الهدف هو تقليل المتوسط المرجح لتقليل عدد الرموز المستخدمة في الرسالة. ال توجد لطول كلمة المرور لكنه لم يعد كافياً خوارزمية معروفة لحل هذه المشكلة بنفس الطريقة أو بنفس كفاءة ترميز huffmanالتقليدي Coding Huffman تشفير هوفمان

الانتقال إلى للبحث

شجرة هوفمان التي تم إنشاؤها من الترددات الدقيقة للنص "هذا مثال لشجرة هوفمان". ترددات ورموز كل حرف أدناه. يتطلب ترميز الجملة باستخدام هذا الرمز ١٣٥ أو (١٤٧) بت ، مقابل ٢٨٨ أو ١٨٠(بت إذا تم استخدام ٣٦ حرفًا من ٨) أو ٥(بت.)يفترض هذا أن بنية شجرة الكود معروفة لوحدة فك الترميز وبالتالي ال يلزم احتسابها كجزء من المعلومات المرسلة.

Char Freq Code

١١١ ٧ مسافة

A 4 010

E 4 000

F 3 1101

H 2 1010

I 2 1000

M 2 0111

N 2 0010

S 2 1011

T 2 0110

L 1 11001

O 1 00110

P 1 10011

R 1 11000

U 1 00111

X 1 10010

يمكن عرض الإخراج من خوارزمية هوفمان كمتغير - جدول كود الطول لتشفير رمز المصدر مثل حرف في ملف. تستمد الخوارزمية هذا الجدول من الاحتمال المقدر أو تكرار الحدوث (الوزن) لكل قيمة محتملة لرمز المصدر. كما هو الحال في طرق ترميز اإلنتروبيا الأخرى ، يتم تمثيل الرموز الأكثر شيو ًعا بشكل عام باستخدام وحدات بت أقل من الرموز الأقل شيو ًعا. يمكن تنفيذ طريقة هوفمان بكفاءة ، وإيجاد رمز في الوقت المناسب خطي إلى عدد أوزان الإدخال إذا

تم فرز هذه الأوزان. ومع ذلك ، على الرغم من أن ترميز هوفمان هو الأمثل بين طرق ترميز الرموز بشكل منفصل ، إال أنه ليس داء ًما أفضل بين جميع طرق الضغط - يتم استبداله بـ ترميز حسابي أو أنظمة رقمية غير متماثلة إذا مطلوب نسبة ضغط افضل

LZW()

هي خوارزمية مشهورة لضغط البيانات بدون ضياع تم إنشاؤها من ١٩٧٨قبل آبراها ليميل وجاكوب زيف وتيري ويلش. نشرت من قبل فيلش عام ١٩٨٤ تحسينا على خوارزمية lzw المنشورة من قبل ليميل وزيف عام .١٩٧٨ الخوارزمية مصممة لتكون سريعة لكنها عادةً ال تصل للحالة الممثلية من الضغط ألنها تقوم بتحليل محدود للبيانات. ،% طبعاً باستخدام خوارزمية lzw تم التخلص من ٦ بتات، أي أن النص قد تم ضغطه بنسبة ١٢

هذه النسبة ضئيلة هنا ألن الأبجدية المستخدمة بسيطة ٤ رموز فقط بينما في حال تم استخدام أبجدية أكبر كجدول أسكي مثالً فإن النسبة سترتفع ألن عدد البتات اللازمة للمحرف قبل الترميز سيكون أكبر من عدد البتات اللازمة للقيمة

الفصل الثاني: الجزء العملي

مقاييس البحث واجزاء البحث

في هذا البحث تم إجراء العمل على تجربتين كل تجربة تحتوي على ثلاث اختبارات التجربة الأولى تتضمن إجراء الاختبارات باستخدام طريقة LZW

وأما التجربة الثانية فتم استخدام طريقة Haffman..تم إجراء ثلاث اختبارات في كل تجربة على ملفات ذات أحجام مختلفة كما موضح في الجدول(٣-١)

ملفات الاختبار

اسم الملف	حجم الملف قبل الضغط	حجم الملف بعد الضغط
File1	١كيلوبايت	
File2	١٠كيلو بايت	
File3	١٠٠ كيلو بايت	

تم حساب مجموعة من المقاييس والمعايير لطرق تقسيم نتائج العمل .

نسبة الضغط $\boldsymbol{Percentage\ Saving = size\ of\ input\ stream - size\ of\ output\ strea}$

$$\boldsymbol{\div stream\ input\ the\ of\ size}$$

×%

****

$$\boldsymbol{Factor\ Compression}$$

$$=$$

$$\boldsymbol{strea\ input\ the\ of\ size}$$

÷

$$\boldsymbol{stream\ output\ the\ of\ size}$$

.....

$$\boldsymbol{pixel\ per\ Bit}$$

$$=$$

$$\boldsymbol{(file\ compressed)Bits\ of\ Number)}$$

$$\boldsymbol{(file\ original)Pixels\ of\ Number)}$$

وكانت مواصفات الحاسبة التي تم إجراء الاختبارات فيها هي كالتالي :

المواصفات الفنية للحاسبة Dell

DDR3 ذاكرة ٢GB 1066MHz

من الصعب • ٥٠٠ غيغابايت، (5400RPM) SATA القرص

عرض • ١٤ "(768 × 1366) LED HD مزودة بتقنية TrueLife

الرسومات • الرسومات HD إنتل

الصوت • صوت عالي الوضوح

• ٢ مكبرات صوت ستيريو مدمجة

• ميكروفون المتكاملة

• محرك الأقراص الضوئية DVD-سوبيرمولتي سرعة فائقة (DVD ± RW)

• ١٠٠/١٠ ميغابايت في الثانية شبكة إيثرنت

شبكة لاسلكية • بمعيار ٨٠٢,١١g / ن

• ١ × منافذ VGA

• ١ × الأحمر

• ٣ × USB 2.0

المخرجات • ١ × سماعة الصوت

• ١ × إدخال الصوت ميكروفون

قارئ بطاقة الذاكرة الرقمية الآمنة • (SD)، الآمنة عالية السعة الرقمية (SDHC)، وبطاقات الوسائط المتعددة (MMC) و Memory Stick (MS)، ذاكرة للمحترفين (MS-برو).

• كاميرا ٠,٣ ميجا بكسل

• ماوس لوحة اللمس المتكاملة

• بطارية ليثيوم ٦ خلايا

• محول ٦٥ واط

لا يشمل • نظام التشغيل نظام التشغيل (تباع على حدة)

الأبعاد • ٣٤ س ٢,٣ / ٣,٥ × ٢٤,٥CM (م خ ح خ د)

نتائج التجربة الأولى

في هذه التجربة تم إجراء ثلاث اختبارات على ملفات بأحجام مختلفة وكانت النتائج بعد الضغط باستخدام طريقة LZWهي إنقاص حجم الملفات الأصلية بنسب مختلفة كما موضح في الجدول **(٣-٢)**

اسم الملف	حجم الملف قبل الضغط	حجم الملف بعد الضغط
File1	١كيلو بايت	
File2	١٠كيلوبايت	
File3	١٠٠كيلوبايت	

المصادر

- **^** Wade, Graham (1994). *Signal coding and processing* (2 ed.). Cambridge University Press. p. 34. *ISBN 978-0-521-42336-6. Retrieved 2011-12-22*. The broad objective of source coding is to exploit or remove 'inefficient' redundancy in the *PCM* source and thereby achieve a reduction in the overall source rate R.
- **^** Mahdi, O.A.; Mohammed, M.A.; Mohamed, A.J. (November 2012). *"Implementing a Novel Approach an Convert Audio Compression to Text Coding via Hybrid Technique" (PDF)*. International Journal of Computer Science Issues. **9** (6, No. 3): 53–59*. Retrieved 6 March 2013*.
- **^** Pujar, J.H.; Kadlaskar, L.M. (May 2010). *"A New Lossless Method of Image Compression and Decompression Using Huffman Coding Techniques" (PDF)*. Journal of Theoretical and Applied Information Technology. **15** (1): 18–23.
- **^** Salomon, David (2008). A Concise Introduction to Data Compression. Berlin: Springer. *ISBN 9781848000728*.
- **^** S. Mittal; J. Vetter (2015), "A Survey Of Architectural Approaches for Data Compression in Cache and Main Memory Systems", IEEE Transactions on Parallel and Distributed Systems (IEEE) **27** (5): 1524–1536, *doi*:*10.1109/TPDS.2015.2435788*
- **^** Tank, M.K. (2011). "Implementation of Lempel-ZIV algorithm for lossless compression using VHDL". Implementation of Limpel-Ziv algorithm for lossless compression using VHDL. Berlin: Springer. pp. 275–283. *doi*:*10.1007/978-81-8489-989-4_51*. *ISBN 978-81-8489-988-7*
- *https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=*
- **https://www.google.com/url?sa=t&rct=j&q=&esrc=s**
- https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web

مصادر باللغه العربية

- مركز الاحصاء ، مفاهيم عامة حول البيانات الكبيرة ، أدلة المنهجية والجودة ، دليل رقم ١٣ ابو طبي صفحه ٤
- حكومة دولة الامارات العربية المتحدة ٢٠٢٠تقرير فرص المستقبل