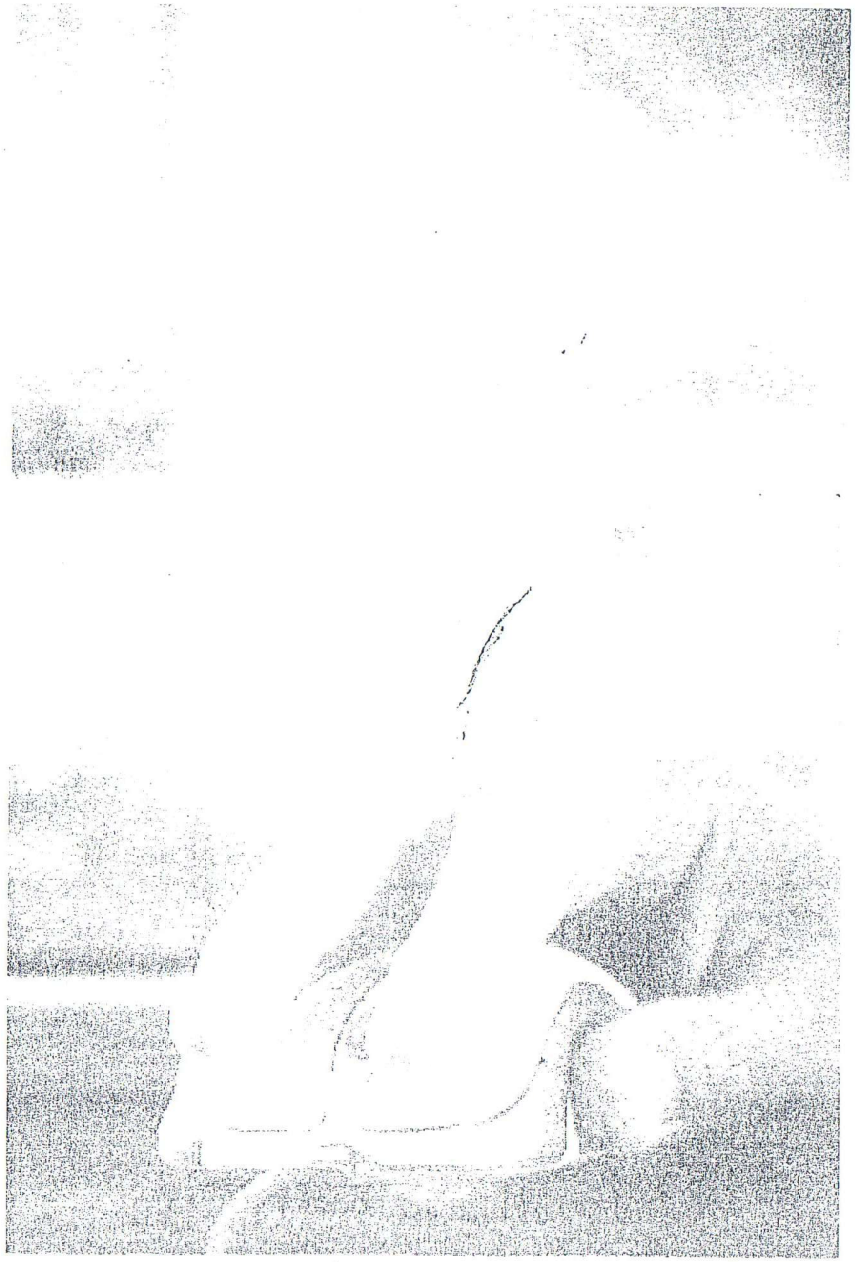
الوحدة الثالثة

الصورة الرقمية ومعالجتها

Digital Image Processing



لا بد أنك قد شاهدت صورة عبر جهاز الحاسوب، ولا بد أنك قد لاحظت اختلافها، ولو بعض الشيء عن تلك التي تحصل عليها عند تحميض الأفلام المصورة بالكاميرا العادية. تحاول هذه الوحدة أن تشرح لك أسباب هذا الاختلاف، مزاياه وعيوبه. كما تعرض الوحدة لطرق إدخال الصورة وعرضها، ويشمل ذلك الصور ذات المصادر الخارجية كالكاميرات الرقمية أو العادية.

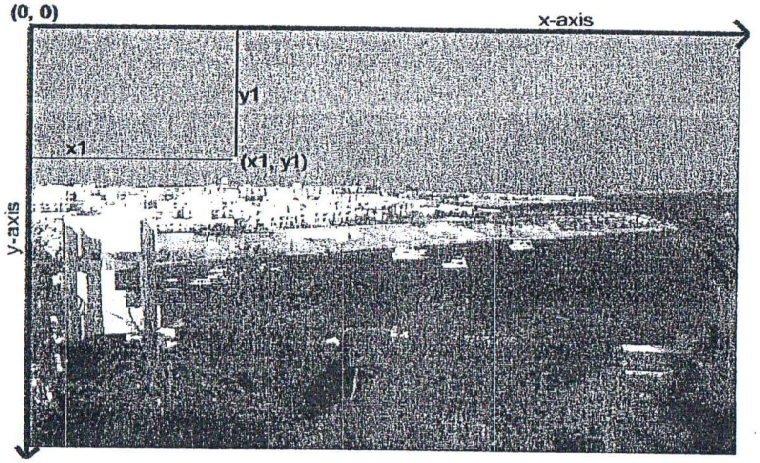
ويبقى أهم ما أدخله الحاسوب إلى عالم الصور هو إمكانية معالجة الصور وتعديلها أو تغييرها. فأصبح بإمكانك اليوم إنتاج صورة رقمية تظهر فيها أمام برج إيفل أو تاج محل أو جسر لندن! كل ذلك باستخدام طرق معالجة الصور المختلفة. كما تتيح لك هذه الوحدة إمكانية اختبار مثل هذه الوسائل المختلفة لمعالجة الصور باستخدام برمجيات متوفرة على أجهزة الحاسوب الشخصية مثل برنامج Photoshop. وقد تهدف معالجة الصور أحيانا إلى تعديل الصورة لإزالة شوائب فيها وتحسين نوعيتها. مثل هذه الطرق تجد الكثير من التطبيقات في تعديل الصور الفوتوغرافية خصوصا القديمة منها والتي أثرت عوامل الزمن سلبا على نوعيتها.

1. تمثيل بيانات الصورة الرقمية

1.1. بيكسل (نقطة ضوئية) وإحداثياتها

تقسم الصورة – كما هو الحال في شاشة التلفزة أيضا– إلى مجموعة من النقاط الضوئية والمسماة بيكسل Pixel. يمكن ملاحظة هذه النقاط المضيئة عند الاقتراب بشكل كاف من هذه الشاشة. لكن مشاهدة الشاشة عن بعد يؤدي إلى دمج هذه النقاط معاً في دماغ الإنسان لتظهر كأنها نسيج واحد يشكل الصورة الضوئية. لذا فإن العقل البشري لا يميز بين هذه النقاط بل يجمعها معا ليشكل المشهد المرئي.

يقوم جهاز الحاسوب بتخزين الصورة وعرضها اعتماداً على البيكسل. حيث تخزن الصورة على شكل مجموعة من النقاط كما يبين الشكل 1.3:



شكل 1.3 التمثيل النقطي للصورة

يكون الطرف العلوي الأيسر من الصورة هو نقطة الانطلاق حسب الفهم الحوسبي للصورة. حيث يمثل محور السينات بالحد العلوي للصورة ومحور الصادات بالحد الأيسر لها كما هو مبين في الشكل 1.3. وعليه فإن البيكسل المبين يتم ترميزه حسب بعده العمودي (الصادي) والأفقي (السيني) عن نقطة الأصل (0، 0). وهكذا فإن الصورة تكون عبارة عن مجموعة من البيكسلات الصغيرة التي تكون الصورة أفقياً ورأسياً. وتمتد هذه البيكسلات الصغيرة لتشمل الصورة كلها.

2.1. التمايز Resolution

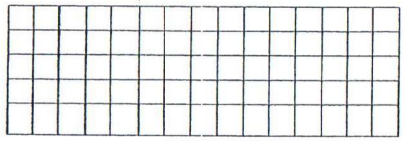
يعرّف التمايز Image Resolution بأنه درجة الدقة في تمثيل الصورة، حيث يمثل التمايز بعدد النقاط الضوئية أو عدد البيكسلات في الصورة. فكلما زاد عدد البيكسلات زادت الدقة في الصورة وبالتالي زاد التمايز.

كما يعتمد التمايز في عرض الصورة على إمكانيات الشاشة أو جهاز العرض المقدم. فتتيح الشاشات المستخدمة في أجهزة الحاسوب الشخصية PC إمكانية تعديل التمايز في الشاشة بزيادة أو تقليل عدد البيكسلات فيها. ومن أمثلة التمايز المستخدمة في الشاشات: 800×600، 1024×768، 1280×960. وكما هو واضح من هذه الأرقام أن التمايز 800×600 يتكون من 480000 نقطة مضيئة على الشاشة في حين يوفر التمايز 1280×960 ما مقداره 1228800 نقطة مضيئة على الشاشة!

واعتماداً على تمثيل الشاشة وتمايز الصورة فإننا نعرض الصورة الموجودة أمامنا. فإذا تكونت الصورة من 100 بيكسل مضيئة أفقياً و100 بيكسل مضيئة عمودياً فإنها تكون في المجموع الكلي مكونة من 10000 بيكسلاً. ويمثل تمايز الصورة عادة بطريقة: 100×100، حيث يبين هذا التمثيل عدد البيكسلات العمودية والأفقية وبالتالي تمايز الصورة الكلي.

مثال

يظهر الشكل التالي عدد البيكسلات في الصورة. ما هو تمايز هذه الصورة؟

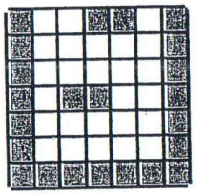


الحل:

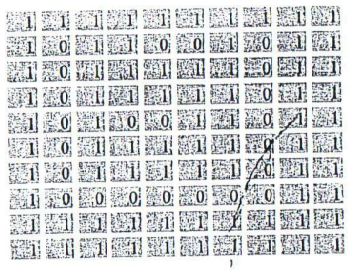
تتكون الصورة من 15 بيكسلاً أفقياً و 5 بيكسلات عمودية. أي أنها تحتوي 75 بيكسلاً. تمايز هذه الصورة هو 15×5.

ويقوم الحاسوب بتخزين معلومات تتعلق بسباللون حول كل بيكسل. فلنأخذ على سبيل المثال الصور التي تمثل فيها البيانات بالأبيض والأسود فقط، ولنقترح أن الأسود يمكن تمثيله بالرمز صفر (0) في حين يمثل اللون الأبيض بالرمز واحد (1). في هذه

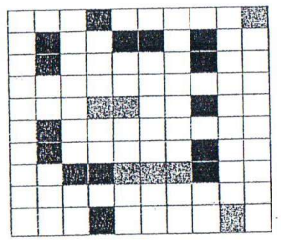
الحالة فإن الحاسوب يخزن الرقم صفر عند كل بيكسل أسود، والرقم واحد عند كل بيكسل أبيض كما في المثال التالي:



الصورة أعلاه مكونة من 100 بيسكسل (10×10)، وبالتالي يمكن تخزين المعلومات حول كل بيكسل كما يلي :



وهكذا فإن كل بيكسل في المثال السابق يمثل اللون الأبيض أو الأسود بـbit واحد فقط. أما لو استخدمنا 2 bit لتمثيل كل بيكسل فإننا نصبح أمام أربع درجات مختلفة كما في المثال التالي:



وباستخدام 2 bits، يمكن تمثيل اللون الأبيض ب 11، واللون الرمادي الفاتح 10، واللون الرمادي الغامق بـ 01 بينما يمثل الأسود بالرمز 00. وبالتالي يمكن تخزين المعلومات حول كل بيكسل كما يلي:

11	11	11	01	11	11	11	11	11	10
11	00	11	11	00	00	11	00	11	11
11	01	11	11	11	11	11	00	11	11
11	11	11	11	11	11	11	11	11	11
11	11	11	10	10	11	11	00	11	11
11	01	11	11	11	11	11	11	11	11
11	01	11	11	11	11	11	00	11	11
11	11	01	01	10	10	10	00	11	11
11	11	11	11	11	11	11	11	11	11
11	11	11	00	11	11	11	11	10	11

ويسمى عدد الbits المستخدمة في تمثيل اللون في الصورة بالتمايز اللوني Color Resolution. وهكذا يمكنك أن تميز بين نوعين من التمايز: تمايز الصورة وهو عدد البيكسلات في الصورة، والتمايز اللوني وهو عدد الbits المستخدمة في تمثيل لون كل بيكسل في الصورة. وبزيادة التمايز اللوني فإنه يمكن تمثيل عدد أكبر من الألوان. فإذا استخدمنا 8 bits مثلاً في تمثيل اللون فإنه يصبح بإمكاننا تمثيل 256 لوناً مختلفاً. ويسمى هذا التمثيل (8 bits لكل بيكسل)، بالتدرج الرمادي Gray Scale. وإذا استخدمنا 24 bits فإننا نصبح قادرين على أن نمثل 16777216 لوناً مختلفاً، وهكذا.

أي أن عدد الألوان = $2^{\text{التمايز اللوني}}$

مثال

إذا كان التمايز اللوني في الصورة هو 6 bits فما هو عدد الألوان التي يمكن تمثيلها في تلك الصورة؟

الحل:

التمايز اللوني هو عدد bits المستخدمة لتمثيل اللون في كل بيكسل. فالعدد 6 يرمز إلى 6 bits/pixel. وباستخدام 6 bits فإنه يمكننا تمثيل 2^6 قيمة مختلفة وتساوي 64 قيمة مختلفة، أي 64 لوناً مختلفاً.

ويسهل الاستنتاج أن التمايز بنوعيه: تمايز الصورة والتمايز اللوني، يحدد الحجم التخزيني للصورة. فكلما زاد عدد البيكسلات في الصورة، احتاجت مساحة تخزينية أكبر. وكلما زاد عدد الbits المستخدمة في تمثيل كل بيكسل، زادت بالطبيعة المساحة التخزينية للصورة.

حجم الصورة = تمايز الصورة x التمايز اللوني Image File Size = Image Resolution x Color Resolution

مثال:

صورة مكونة من 100×100 بيكسل، واستخدم التدرج الرمادي (8 bits) في تمثيل كل بيكسل، إحسب حجم الصورة.

الحل:

حجم الصورة $= 100\times100\times8=80000\ bits=80\ Kbit=10\ KB$

تذكر:

1Byte (B) = 8 bits (b)
1000b = 1 Kb
1000Kb = 1 Mb

وأحيانًا تحسب

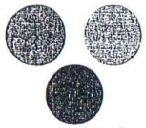
1024b = 1 Kb

2. الفضاء اللوني

تحدثنا في القسم السابق عن تمثيل الصورة بالألوان: الأبيض، الأسود والرمادي. ولكن الصور الملونة تبقى هي الأوسع انتشاراً في الصورة الحوسبة. ولا زال البحث مستمراً عن التمثيل الأنسب للألوان. ذلك أن الألوان الحقيقية في العالم لا حصر لها، لكن الحاسوب يحاول فقط تمثيل أكبر قدر من هذه الألوان بشكل يجعل من الممكن لنا أن نرى الصور الحاسوبية ونشعر أنها صور واقعية.

وقد وضع العلماء عدة أنظمة لتمثيل اللون في الحاسوب. لعل أشهر هذه الأنظمة هو ذاك المستمد من تمثيل الألوان في التلفاز والسينما. ذلك أن التلفاز يمثل اللون على شكل مزيج من ألوان ثلاثة رئيسة هي: الأحمر، الأخضر والأزرق. ولذا كان من الطبيعي أن يحاول علماء الحاسوب اقتباس هذا التقسيم التلفزي وتطبيقه. وقد سمي هذا الفضاء اللوني (RGB Red) Green, Blue.

لكن العلماء لم يكتفوا بهذا الفضاء اللوني وحده، وحاول كثير منهم استكشاف فضاءات لونية أخرى وطرق أخرى للتمثيل اللوني. ذلك أن التمثيل اللوني: (أحمر، أخضر، أزرق) ليس مستمداً من فهم الإنسان للألوان، وإنما هو تمثيل من ابتكار الإنسان ليس إلا. يقوم هذا القسم من الوحدة باستعراض الفضاءات اللونية شائعة الاستخدام بشكل تفصيلي.



1.2. RGB

كما أوضحنا في المقدمة سالفة الذكر، فإن الفضاء اللوني المسمى RGB مشتق من التمثيل التلفزي للألوان.

في الفضاء اللوني RGB يستخدم 24 bits لتمثيل اللون في كل بيكسل من الصورة. فإذا أخذنا بيكسلاً واحداً فإننا قد نجد تمثيل اللون لهذا البيكسل على الشكل:

110110010011101100001111

ولفهم هذا التمثيل فإنه يجب أن نعلم أن كل 8 bits تمثل أحد الألوان الثلاثة الرئيسة: الأحمر، الأخضر والأزرق. وهكذا فإنه يمكن تقسيم هذا التمثيل إلى ثلاثة أقسام كما يلي

11011001	00111011	00001111
أحمر Red	أخضرGreen	أزرق Blue

وبما أن كل لون يمثل ب 8 bits فإننا نستنتج أن درجة اللون قد تتراوح بين 0 و 255 لكل من الألوان الثلاثة (2^8).

وقد شاع تمثيل الفضاء اللوني RGB عن طريق النظام السادس عشر Hexadecimal خصوصا في استخدامات الألوان على صفحات الويب. ويمثل كل لون من هذه الألوان بالنظام السادس عشر كالآتي:

1. يقسم كل 8 bits ممثلة للون إلى قسمين يكون كل منهم 4 bits.
2. يحول كل 4 bits إلى النظام السادس عشر حسب الجدول 1.3:

النظام العشري	النظام الثنائي	النظام السادس عشر
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

جدول 3. 1 النظام السادس عشر Hexadecimal

3. يتم جمع التمثيل السادس عشر للألوان الثلاثة حسب الترتيب: أحمر، أخضر، أزرق.
4. يضاف الرمز (#) أو (OX) للدلالة على التمثيل باستخدام النظام السادس عشر.

مثال

وضح التمثيل السادس عشر للون 110110010011101100001111

الحل:

1. تقسم كل 8 bits إلى مجموعتين 4 bits لكل منها	00001111		00111011		11011001	
	1111	0000	1011	0011	1001	1101
2. يتم تحويل كل مجموعة إلى مقابلها في النظام السادس عشر	F	0	B	3	9	D
3. يتم تجميع المجموعات معا	F0		B3		D9	
4. إضافة رمز # إلى بداية التمثيل	#D93B0F					

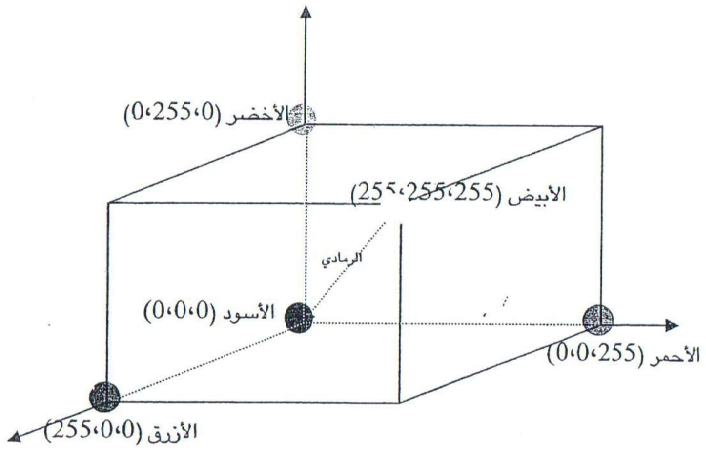
تدريب (3)

حول قيمة البيكسل التالية بنظام RGB إلى التمثيل السادس عشر:
101101011000001001101010

تدريب (4)

حول التمثيل السادس عشر بنظام RGB إلى قيم الألوان الثلاثة الرئيسة: الأحمر، الأخضر والأزرق بالنظام العشري: #CC76F3

ويمثل الفضاء اللوني RGB على شكل مكعب (ثلاثي الأبعاد) كما في الشكل 3.3:



شكل 3. 2 الفضاء اللوني RGB

يوضح الشكل 2.3 كيفية تمثيل الألوان الأساسية إضافة إلى اللونين الأسود والأبيض بنظام RGB كما في الجدول 2.3:

اللون	الأحمر	الأخضر	الأزرق
الأحمر	255	0	0
الأخضر	0	255	0
الأزرق	0	0	255
الأبيض	255	255	255
الأسود	0	0	0

جدول 3. 2 الألوان الأساسية حسب الفضاء اللوني RGB

كما يبين الشكل كيف أن خليطاً متساوياً من الألوان الثلاثة يؤدي دائما إلى اللون الرمادي، وكلما زادت نسبة الألوان في الخليط فإن اللون يصبح فاتحاً أكثر وصولاً إلى اللون الأبيض. كما في الجدول 3.3:

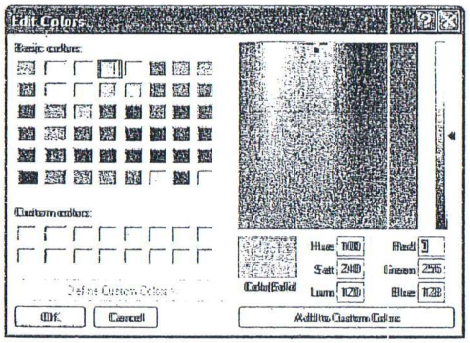
اللون	الأحمر	الأخضر	الأزرق
	0	0	0
	50	50	50
	100	100	100
	150	150	150
	200	200	200
	255	255	255

جدول 3. 3 درجات اللون الرمادي حسب الفضاء اللوني RGB

ولعلك تتساءل إن كان بإمكاننا فعلاً تمثيل جميع الألوان التي تتخيلها بتلك الألوان الثلاثة الرئيسة: الأحمر، الأخضر والأزرق. وللإجابة عن ذلك التساؤل سنأخذ بعض الأمثلة:

اللون الأصفر: يعبر عن اللون الأصفر بخليط مماثل من اللونين الأحمر والأخضر.

ولتأكيد ذلك استخدم أي تطبيق برمجي للصور مثل Paint أو Photoshop. قم بفتح هذا التطبيق وافتح النافذة الخاصة بالألوان كما في الشكل 3.3



شكل 3. 3 نافذة الألوان في برمجية Paint

ستجد الفضاء اللوني RGB ممثلاً بالألوان الثلاثة الأساسية. استبدل الفراغات المقابلة للألوان بما ستجده في الجدول التالي للحصول على الألوان:

اللون	الأحمر	الأخضر	الأزرق
الأصفر	255	255	0
الفسفوري	5	250	61
البنفسجي	90	20	220

ويمكنك أيضاً أن تحاول إدخال أية قيم لونية تريدها وتلاحظ الألوان الناجمة عن الخلط.

ولا يجب أن يسود الاعتقاد أن هذه القيم تقاس فقط بالتجربة، بل إن التفكير في هذه القيم يقودنا إلى استنتاج اللون أيضا. فاللون البرتقالي مثلا هو خليط من الأحمر والأصفر (جرب خلط الألوان المائية!) والأصفر بدوره خليط من الأحمر والأخضر، وبالتالي فإنه يمكن إنتاج اللون البرتقالي بخلط كمية أكبر من الأحمر من كمية اللون الأخضر. وبالتالي يمكن تمثيل اللون البرتقالي بالقيم التالية مثلا: الأحمر: 255، الأخضر: 128، الأزرق: 0. استخدم التطبيق اللوني السابق في إثبات ذلك أيضاً.

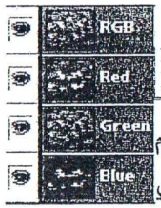
تدريب (5)

اللون	الأحمر	الأخضر	الأزرق
؟؟؟	0	255	255
؟؟؟؟	200	50	200

فكر في التركيبة التالية من الألوان وحاول تقدير اللون الناتج، ثم استخدم تطبيقاً للصور في اكتشاف ما إذا كان ما فكرت به صحيحاً:

نشاط

يمكن دراسة الصور الملونة حسب الفضاء اللوني RGB باستخدام تطبيق الصور Photoshop، وذلك باستخلاص اللون الأحمر فقط في



الصورة، أو الأخضر أو الأزرق. حيث يمثل كل لون قناة Channel في الصورة.

افتح صورة ملونة باستخدام تطبيق الصور ثم طبق الأمر: Window -> Channels من القائمة في أعلى التطبيق

بإمكانك إخفاء أو إظهار قناة معينة بالضغط على رمز العين حيث يمكنك التركيز على قناة لونية معينة أو أكثر في كل مرة. تمثل كل قناة من القنوات الثلاثة نسبة اللون الأحمر/الأخضر/الأزرق في الصورة.

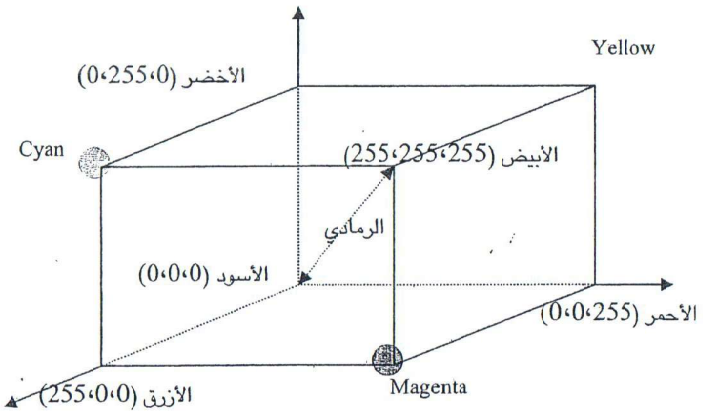
2.2. CMYK

سبق وأخبرناك أن الفضاء اللوني (أحمر، أخضر، أزرق) هو أشهر الفضاءات اللونية وأكثرها شيوعاً نظراً لارتباطه بالتمثيل اللوني على شاشة التلفاز، إلا أنه بالطبع ليس الفضاء اللوني الوحيد في عالم الحاسوب.

وهناك فضاء لوني آخر ذو استخدام واسع عندما يرد المرء طباعة الصور الملونة. فعندما تطبع الصور الملونة ذات الفضاء اللوني RGB فإن اللون الأسود لا يحتوي أي مكونات من الأحمر أو الأخضر أو الأزرق (0، 0، 0). وعليه لا يتم دمج أية كميات من هذه الأحبار الملونة لطباعة اللون الأسود!! إذ يفترض هذا الفضاء اللوني أن اللون الأسود هو عدم وجود لون على الإطلاق، أي أن الأوراق المستخدمة في الطباعة يجب أن تكون سوداء اللون، وبما أن الحال يختلف في الواقع عن هذا، ذلك أن اللون الأبيض هو لون ورق الطباعة، وهو ما لا يجب استخدام أي كميات من الحبر في تشكيله، فإن الفضاء اللوني RGB ليس صالحاً لهذا الاستخدام. لحل هذه المعضلة، طور العلماء فضاء لونياً آخر يستخدم اللون الأبيض كأساس وهو الفضاء اللوني CMY اختصاراً للكلمات Cyan, Magenta, Yellow وتترجم الألوان التالية إلى اللغة العربية كما يلي:

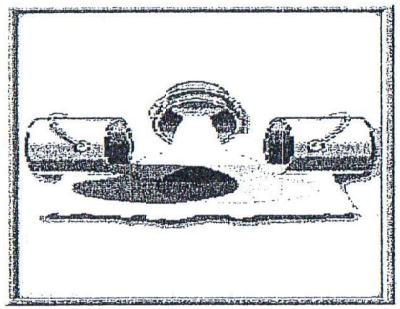
Cyan	أزرق فسفوري
Magenta	الفوشي
Yellow	الأصفر

وتمثل هذه الألوان الثلاثة زوايا المكعب المرسوم في الشكل 3.3 (الزوايا الثلاثة الباقية بعيداً عن زوايا الأحمر والأخضر والأزرق).



شكل 3. 4 مواقع الألوان: Cyan, Magenta, Yellow على مكعب ألوان RGB

في الفضاء اللوني CMY فإن اللون الأبيض يمثل (0،0،0) في حين يمثل اللون الأسود (255،255،255). أي ما يناسب الطباعة وأحبارها. يمثل الشكل 5.3 كيفية دمج الألوان للحصول على الألوان المختلفة بما في ذلك الألوان الأحمر والأخضر والأزرق:



شكل 3. 5 استخدام الألوان Cyan, Magenta, Yellow في الطباعة

كما يمثل الجدول 4.3 قيم بعض الألوان حسب الفضائين اللونيين RGB وCMY:

اسم اللون	الفضاء اللوني RGB			اللون	الفضاء اللوني CMY		
	الأحمر	الأخضر	الأزرق		Cyan	Magenta	Yellow
الأبيض	255	255	255		0	0	0
الأسود	0	0	0		255	255	255
الأحمر	255	0	0		0	255	255
الأخضر	0	255	0		255	0	255
الأزرق	0	0	255		255	255	0
Cyan	0	255	255		255	0	0
Magenta	255	0	255		0	255	0
الأصفر	255	255	0		0	0	255

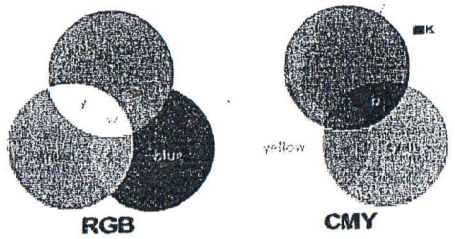
جدول 3. 4 مقارنة الألوان الأساسية بين الفضائين اللونيين RGB وCMY

قم بدراسة الجدول السابق بدقة... يمكنك أن تستنتج المعادلات التالية والمستخدمة للتحويل من الفضاء اللوني RGB إلى الفضاء اللوني CMY من الجدول وتحاول فهمها بتعمق:

$$C = 255 - R$$
$$M = 255 - G$$
$$Y = 255 - B$$

إلا أن الفضاء اللوني CMY واجه صعوبة تقنية في التطبيق، فمن المعروف أن النظرية تختلف حقيقة عن الواقع التطبيقي. ففي الاستخدام، لوحظ أن اللون الناتج عن دمج كميات متساوية (قدر الإمكان) من الأزرق الفسفوري والفوشي والأصفر ينتج لوناً بنياً داكناً وليس أسود نقياً. ذلك أن تحقيق كميات متساوية تماماً من الألوان الثلاثة هو أمر أقرب إلى المستحيل.

عالج العلماء الفجوة بين النظرية والتطبيق في الفضاء اللوني CMY بإصدار نسخة محسنة من هذا الفضاء اللوني أطلق عليها اسم الفضاء اللوني CMYK اختصاراً للألوان Cyan, Magenta, Yellow, Black. ونلاحظ أن هذا الفضاء اللوني هو مجرد إضافة للون الأسود. ويمثل اللون الأسود الناصع في هذا الفضاء المحسن بالرموز (0، 0، 0، 100٪). أي أن اللون الأسود أصبح يشكل من حبر أسود اللون بدل تشكيله من دمج الأحبار المكونة للألوان الثلاثة: الأزرق الفسفوري، الفوشي والأصفر كما يبين الشكل 6.3.



شكل 6.3 الفروق بين نظامي RGB وCMY

ويمكن احتساب قيم CMYK من قيم CMY كما يلي:

أولاً: نقوم بتحديد أقل قيمة بين القيم الثلاث C، M، Y ولنسمّ هذه القيمة L

ثانيا: تحتسب قيم CMYK كما يلي:

$$C = \frac{C - L}{255 - L}$$

$$M = \frac{M - L}{255 - L}$$

$$Y = \frac{Y - L}{255 - L}$$

$$K = \frac{L}{255}$$

ثالثاً: يتم إيجاد النسب المئوية للأرقام بالتقريب.

مثال:

حول القيم (200، 134، 96) من النظام RGB إلى النظام CMYK

الحل:

1. نقوم بتحويل القيم من الفضاء اللوني RGB إلى CMY كما يلي

$$C = 255 - R = 255 - 96 = 159$$

$$M = 255 - G = 255 - 134 = 121$$

$$Y = 255 - B = 255 - 200 = 55$$

وهكذا CMY = (159، 121، 55)

2. نقوم بعد ذلك بالتحويل بين النظام CMY وCMYK

أقل قيمة = 55

$$C = \frac{C - L}{255 - L} = \frac{159 - 55}{255 - 55} = \frac{104}{200} = 0.52$$

$$M = \frac{M - L}{255 - L} = \frac{121 - 55}{255 - 55} = \frac{66}{200} = 0.33$$

$$Y = \frac{Y - L}{255 - L} = \frac{55 - 55}{255 - 55} = \frac{0}{200} = 0$$

$$K = \frac{L}{255} = \frac{55}{255} \approx 0.216$$

بعد التقريب CMYK = (33٪، 0٪، 21.6٪، 52)

مثال 2:

حول اللون #7AB50F إلى الفضاء اللوني CMY.

الحل:

1. نقوم بتحويل التمثيل السادس عشر لنظام RGB إلى التمثيل العشري

$(7A)_{16} = 7 \times 10 + 10 = 122$ اللون الأحمر

$(B5)_{16} = 11 \times 16 + 5 = 181$ اللون الأخضر

$(F0)_{16} = 0 \times 16 + 15 = 15$ اللون الأزرق

RGB = (122 ،181 ،15)

2. نقوم بتحويل RGB إلى CMY

C = 255 – R = 255 – 122 = 133

M = 255 – G = 255 – 181 = 74

Y = 255 – Y = 255 – 15 = 240

إذن CMY = (133240 ،74 ،)

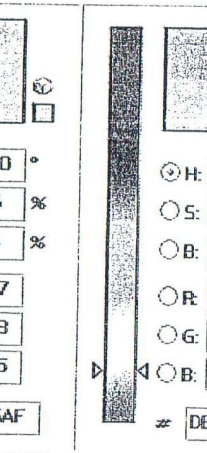
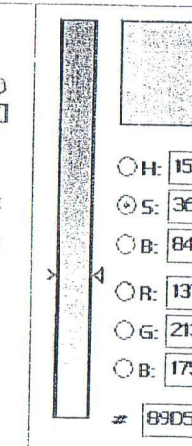
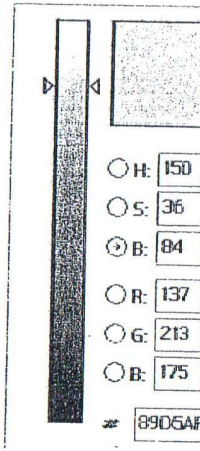
3.2. فضاءات لونية أخرى

ولا تقتصر الفضاءات اللونية على هذين الفضاءين فقط، فهناك المئات من الفضاءات اللونية التي تم إنتاجها لأغراض مختلفة. فمنها ما تم إنتاجه لمعالجة أمور الطباعة كالفضاء اللوني المذكور في القسم السابق CMY، ومنها ما تم إنتاجه لمعالجة الإضاءة المختلفة، حيث ترغب هذه النوعية من الفضاءات اللونية بالحصول على التمثيل اللوني نفسه مهما اختلفت الإضاءة المسلطة عليه، وغيرها من الأهداف التي أنتجت فضاءات لونية متعددة.

ولعل أشهر الفضاءات اللونية التي كانت تسعى إلى فصل عنصر الإضاءة عن التمثيل اللوني هو الفضاء اللوني HSI اختصاراً للكلمات: Hue, Saturation, Intensity. وتترجم هذه الكلمات كالتالي:

Hue	درجة اللون
Saturation	الإشباع اللوني
Intensity	شدة الإضاءة

ويحاول هذا الفضاء اللوني أن يفصل شدة الإضاءة كرقم مستقل، في حين يُبقي الأرقام الممثلة لدرجة اللون والإشباع اللوني ثابتة مهما اختلفت شدة الإضاءة وجدتها. ويمكن فهم هذا الفضاء اللوني بشكل أكثر دقة عن طريق دراسته باستخدام تطبيق Adobe Photoshop:

Hue(H)	Saturation(S)	Intensity(I)
		
أي تغيير في قيمة Hue يؤدي إلى تغيير في اللون نفسه، جميع ألوان الطيف موضحة بقيم Hue المختلفة.	أي تغيير في Saturation يؤدي إلى تغيير في إشباع اللون، ابتداء من اللون غير المشبع تماماً وهو اللون الرمادي، وانتهاء باللون الأخضر المشبع.	أي تغيير في Intensity/ Brightness يؤدي إلى تغيير في الإضاءة بدءاً من اللون الأسود، وهو أقل الألوان إضاءة وانتهاء بالأخضر المضاء إضاءة كاملة.

ويمكن تحويل الفضاءات اللونية من واحد إلى آخر باستخدام معادلات معرَّفة من قبل العلماء. ويمكن ذكر فضاءات لونية أخرى مثل الفضاء اللوني CIE، وهو من أقدم الفضاءات اللونية حيث تم تعريفه عام 1931، أي تم تعريفه لأغراض غير حاسوبية قبل استخدامه في عالم الحوسبة، والفضاء اللوني YUV المستمد من عالم الكاميرات والتلفاز أيضاً.

ففي الفضاء اللوني YUV مثلاً تستخدم المعادلات التالية للتحويل بين نظامي RGB وYUV كما يلي:

$$Y = 0.3R + 0.6G + 0.1B$$

$$U = B - Y$$

$$V = R - Y$$

إلا أن كل فضاء لوني يكون قادراً على تمثيل مجموعة مختلفة من الألوان، ويمكنك إثبات ذلك في النشاط التالي عند تحويل صورة ممثلة بفضاء لوني معين إلى فضاء لوني آخر.

نشاط

قم بفتح صورة ملونة باستخدام تطبيق الصور Photoshop، ثم غير الفضاء اللوني للصورة باستخدام الأمر: Image -> Mode -> CMYK ولاحظ التغير في ألوان الصورة. يمكنك أن تقوم باستخدام أمر الرجوع Undo و Redo عدة مرات حتى تستطيع ملاحظة التغيرات.

3. المسح الضوئي للصور

يفترض أنك قد أصبحت مدركاً لجميع أقسام الصورة الرقمية وتمثيلها، سواء أكانت تستخدم التدرج الرمادي أو أحد الفضاءات اللونية المعروفة. كما تطرقت الأقسام الثلاث السابقة إلى حجم الصورة وتمايزها. إلا أن تمايز الصورة يتأثر بوسائل الإدخال

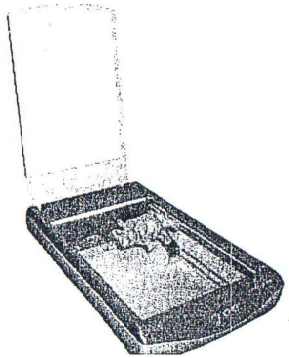
والعرض لهذه الصورة. ذلك أن نسبة ضئيلة من الصور الرقمية هي تلك المنتجة حاسوبياً، في حين أن غالبية الصور الرقمية المتداولة هي تلك المدخلة باستخدام الماسحات الضوئية، الكاميرات الرقمية وغيرها من وسائل الإدخال الرقمية.

إضافة إلى أثر الإدخال على تمايز الصورة ودرجة نقاوتها، فإن أجهزة الإخراج كشاشة الحاسوب أو شاشات العرض المختلفة لها تأثير على كيفية رؤيتنا لهذه الصورة. فإذا كانت هذه الشاشات ذات تمايز يقل عن ذلك الموجود في الصورة نفسها، فإن ذلك التمايز الإضافي الذي تحمله الصورة ليس ذا فائدة كما سنرى في القسم 3.4 من هذه الوحدة.

1.3. الماسحة الضوئية (scanner)

تستخدم عدة أجهزة لإدخال الصور إلى الحاسوب ومعالجتها رقمياً. أحد أشهر هذه الأجهزة هي الماسحات الضوئية scanners، والتي باتت جزءاً أساسياً لا كمالياً من أجهزة الحاسوب الشخصية.

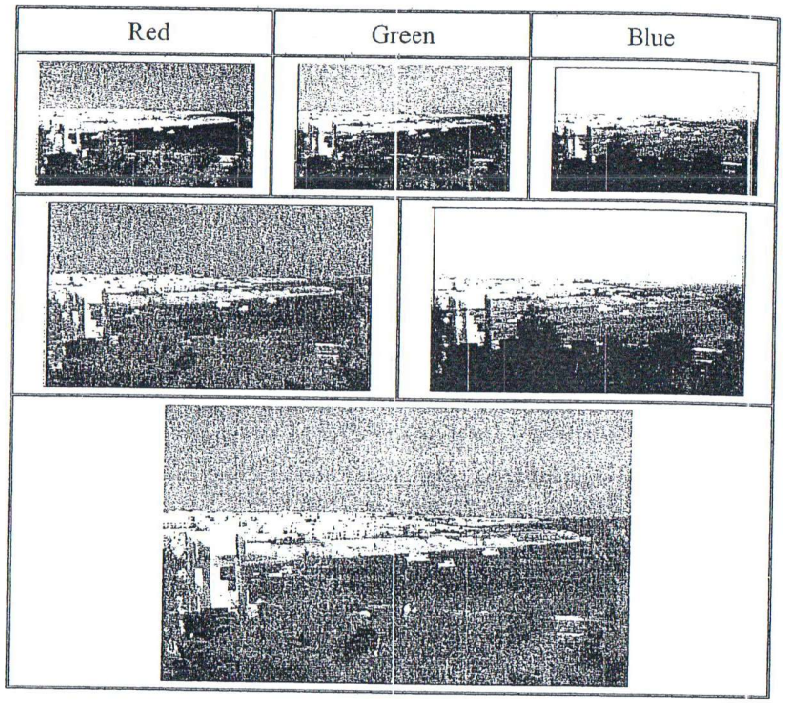
قد يكون مستغرباً أن نقول أن تاريخ الماسحة الضوئية يعود إلى ما قبل الحاسوب! ذلك أن المسح الضوئي كان قد استخدم منذ عام 1926 على يد العالم Robert Ledley، الذي قام باستخدام المسح الضوئي باستخدام أشعة X للحصول على صورة لأجزاء الجسم الداخلية، وما زالت صور الأشعة تستخدم في تشخيص الأمراض. إلا أن الماسحة الضوئية الرقمية كما نعرفها حاليا وحسب الشكل 7.3. تعود إلى عام 1972، وتم إنتاجها على يد مهندس الإلكترونيات البريطاني Godfrey Hounsfield. ولعله يجدر بنا أن نذكر أن الماسحات الضوئية باتت تتخذ أشكالاً أخرى معروفة غير الماسحة الشخصية كتلك المستخدمة في المحلات التجارية، حيث تستخدم الماسحة الضوئية لقراءة الشيفرة الرقمية للمنتجات المباعة والتي تعارفنا عليها باسم Bar Code Scanner وتحويلها إلى رقم رمزي يمثل المنتج المباع دون غيره.



شكل 3. 7 الماسحة الضوئية

عند استخدام الماسحة الضوئية فإننا نقوم بوضع الورقة أو الصورة المراد مسحها على السطح الزجاجي للماسحة الضوئية. تحتوي الماسحة الضوئية ضوءاً داخلياً صغيراً يشع على السطح الزجاجي وما خلفه من معلومات، ثم ينعكس ذلك الضوء على قطعة (CCD Charge-Coupled Device) المشابهة لتلك المستخدمة في الكاميرات الرقمية (كما سنرى لاحقاً). ينتقل هذا الضوء المشع داخل الماسحة الضوئية ليفحص الورقة من أعلاها إلى أسفلها، سطراً سطراً حتى ينتهي من قراءة البيانات الضوئية عنها.

تقوم قطعة CCD بتمييز الألوان الثلاث الرئيسة (الأحمر، الأخضر والأزرق) – راجع الفضاء اللوني RGB في القسم 1.3. – منتجة ثلاث نسخ من الورقة، إحداها للون الأحمر، وأخرى للأخضر، ونسخة ثالثة للون الأزرق. تقوم الماسحة الضوئية بعد ذلك بدمج هذه المكونات الثلاث للون منتجة النسخة النهائية الملونة للورقة أو الجسم المدخل، وتقوم بإرسالها إلى الحاسوب أو الجهاز الرقمي المتصل بالماسحة الضوئية. كما في المثال التالي:



شكل 3. 8 المسح الضوئي للصور
الصورة: شرم الشيخ – مصر (2005)

يمكن فهم المثال في الشكل 8.3 بأخذ النقاط الممثلة للسماء مثلاً كنموذج. حيث يحتوي اللون الأزرق كمية قليلة من الأحمر والأخضر ولذا تبدو هذه المناطق داكنة في المسح الضوئي للونين الأحمر والأخضر (أي قريبة إلى الصفر) في حين تبدو فاتحة (أي ذات قيمة عالية) في المسح الضوئي للون الأزرق.

تختلف الماسحات الضوئية، إحداها عن الأخرى في خاصيتين رئيسيتين: تمايز الماسحة الضوئية، والتمايز اللوني. ويمثل تمايز الماسحة الضوئية عادة برقمين كما يلي: 3200 × 1600 dpi والوحدة المستخدمة للقياس dpi هي اختصار ل (dots per

inch) أي عدد النقاط للإنش الواحد. ففي حين يمثل الرقم الأول – 1600 في المثال السابق- عدد المجسات الضوئية في الإنش الواحد، فإن الرقم الثاني – 3200 في المثال- يوضح عدد المرات التي يتوقف فيها الضوء للقراءة العمودية في الإنش الواحد. أي أن الرقمين يوضحان عدد القراءات الضوئية عموديا وأفقياً في الإنش الواحد. ويعني هذا الرقم أيضاً أن هذه الماسحة الضوئية قادرة على تحقيق تمايز يساوي 3200 × 1600 pixels لكل إنش مربع في الصورة كحد أقصى.

إضافة إلى تمايز الماسحة الضوئية فإن لها تمايزاً لونياً مشابهاً لذلك الذي تم شرحه حول التمايز اللوني للصورة في القسم 2.2. من هذه الوحدة. والتمايز اللوني هو عدد الbits المستخدمة لتمثيل اللون في كل بيكسل في الصورة. وكلما زاد التمايز اللوني فإن عدد الألوان التي يمكن تمثيلها يكون أكبر وبالتالي تكون الماسحة الضوئية أكثر قدرة على تمثيل اللون الطبيعي بشكل أدق. وتتوافر الان ماسحات ضوئية بتمايز لوني قدره 24 bits أو 32 bits أو 48 bits.

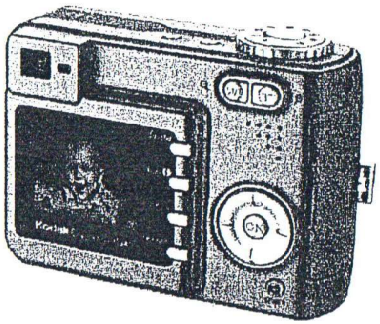
هذا ويجدر ذكر أن بعض الماسحات الضوئية قادرة على إنتاج صور بتمايز أعلى من ذلك الملتقط باستخدام الضوء المشع. يسمى هذا التمايز Interpolated Resolution أي التمايز التوليدي. حيث تقوم البرمجيات المرافقة للماسحات الضوئية بإنتاج هذا التمايز الأعلى عن طريق فحص ومعالجة الصورة المنتجة.

إلا أنه لا ينبغي بالضرورة استخدام الحد الأقصى للماسحة الضوئية في إنتاج الصورة سواءاً كان من حيث تمايز الصورة أو التمايز اللوني. حيث يمكن للمرء اختيار تمايز منخفض لضمان حجم أقل للصورة وبالتالي تقليل الزمن اللازم لاسترجاع هذه الصورة عبر شبكة الإنترنت، راجع القسم 2.2. لدراسة أثر تمايز الصورة والتمايز اللوني على حجم الصورة الكلي.

كما توفر البرمجيات المساندة للماسحات الضوئية إمكانية معالجة الصور البسيطة مثل قص مساحة محددة من الصورة قبيل المسح الضوئي مما يسرّع عملية المسح. حيث

يتم اختيار مساحة معينة من الصورة ويتم المسح الضوئي على تلك المساحة المختارة من الورقة/الصورة فقط بدل المسح الضوئي الكامل، وتسمى هذه العملية Cropping.

2.3. الكاميرات الرقمية



شكل 9.3 الكميرا الرقمية

لا تختلف الكاميرات الرقمية (شكل 9.3) في طريقة عملها كثيراً عن الماسحات الضوئية التي سبق شرح تركيبتها في القسم السابق. فكلاهما تعتمد على (CCD Charge-Coupled Device) في تحويل الضوء المستقبل إلى إشارة كهربية تترجم فيما بعد إلى سلسلة من أرقام الواحد والصفر ليتم تخزينها حوسبياً.

وفي حين تتشابه الكاميرات الرقمية مع الماسحات الضوئية في طريقة عملها، فإنها لا تزال تشبه –وإلى حد كبير– الكاميرات العادية (غير الرقمية) في تركيبها وأجزائها. حيث تتكون الكاميرات الرقمية، على اختلاف أنواعها، من عدسة رئيسة Lens، فتحة لمرور الضوء Aperture، ومصراع الكاميرا Shutter الذي يفتح ويغلق، لفترة زمنية محدودة، ليلتقط الإشارات الضوئية لتسجيل الصورة.

في الكاميرات الرقمية تحديداً، فإنه قبيل أن يفتح مصراع الكاميرا لالتقاط الضوء، يقوم مقياس ضوئي بالتقاط شدة الإضاءة في المشهد ويحدد بناء على ذلك حجم فتحة مرور الضوء Aperture والفترة الزمنية التي يبقى فيها المصراع مفتوحا لالتقاط الضوء. وهكذا فإن إحدى مميزات الكاميرا الرقمية عن تلك العادية أنها تتحكم في إضاءة الصورة بناء على شدة الإضاءة في المشهد مما يقلل احتمالية إنتاج صور ذات إضاءة عالية جداً أو منخفضة للغاية.

بعد ذلك، يفتح مصراع الكاميرا ويمكن للضوء أن يدخل. يسقط الضوء على قطعة CCD التي تحول الإضاءة إلى شحنة كهربية، فكلما ازدادت شدة الإضاءة الساقطة، ازداد بالمقابل مقدار تلك الشحنة الكهربية. ومن ثم تحولت هذه الشحنة إلى نظيرها الخطي.

لا بد هنا أن نذكر أن CCD مسؤولة فقط عن التقاط شدة الإضاءة وليس اللون. تحتوي الكاميرات الرقمية ثلاثة فلاتر مختصة بالألوان الأساسية الثلاث: الأحمر، والأخضر والأزرق –راجع القسم 1.3. من الوحدة– . فالفلتر الأحمر يسمح فقط للطيف الضوئي الأحمر بالمرور عبره، وتقاس شدة الضوء الأحمر المار عبر قطعة CCD. وهكذا هو الحال بالنسبة للونين الأخضر والأزرق أيضاً. عندما يسقط الضوء الأحمر على الفلتر الأحمر في CCD، يتم قياس شدة ذلك اللون (الأحمر) على كل بيكسل، فإذا كانت الإضاءة الحمراء معدومة تسجل القيمة 0، وإذا كانت شديدة تسجل قيماً أعلى تصل في أقصاها إلى 255 للفضاء اللوني RGB وكذلك هو الحال في فلتري اللونين الأخضر والأزرق. ثم تتجمع شدة الإضاءة من الألوان الثلاثة الأساسية لتكوين اللون النهائي للصورة.

تحتوي الكاميرات الرقمية بداخلها معالجاً رقميا processor يقوم بإجراء جميع الحسابات اللازمة لتخزين الصورة، عرضها، وتعديلها أيضا. ويطلق عليه عادة اسم (DSP Digital Signal Processor) أي معالج الإشارة الرقمي. إضافة إلى العمليات الأساسية، يدعم المعالج في مختلف نماذج الكاميرات الرقمية المتوفرة حالياً في الأسواق معظم أشكال الملفات الخاصة بالصور، كملفات JPEG وملفات TIFF. فعلى سبيل المثال تقاس سرعة المعالج المتوافر في معظم كاميرات كوداك بـ8MHz، وهي سرعة تمكنك من التقاط الصورة، ضغطها، تخزينها، نقلها، وعرضها في جزء من الثانية.

كما تقاس جودة الكاميرات الرقمية بقدرتها التخزينية، أي المساحة التي يمكن تخزين الصور بداخلها (وبالتالي عدد الصور الممكن تخزينها). إضافة إلى تمايز الصورة كما

سبق وشرحنا حول الماسحات الضوئية. وتتوافر حالياً كاميرات رقمية بقدرة تمايز تساوي 5 Mega Pixels أو 8 Mega Pixels وبأسعار تسمح باقتنائها على المستوى الشخصي.

مثال:

كاميرا رقمية بمساحة تخزينية مقدارها 8MB، إذا كان تمايز الصورة يساوي 100 × 100 بيكسل باستخدام الفضاء اللوني RGB. احسب أكبر عدد ممكن من الصور الممكن تخزينها داخل هذه الكاميرا

الحل:

حجم الصورة الواحدة =

$$100 \times 100 \times 24 = 240000 \text{ bits} = 30000 \text{ byte} = 30 \text{ KB}$$

عدد الصور = المساحة الكلية / حجم الصورة الواحدة = 8000 KB /30KB

= 266 صورة

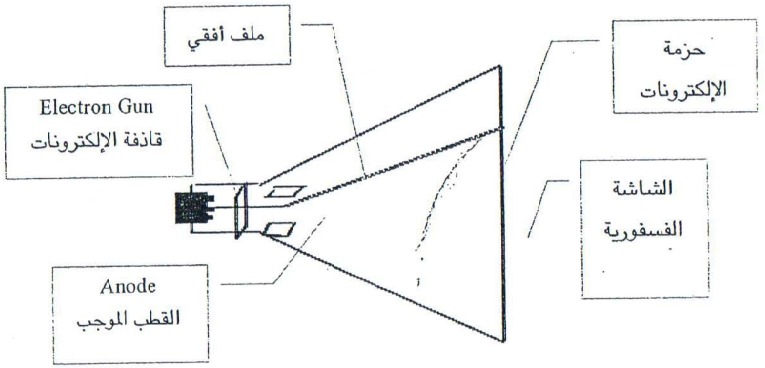
3.3. شاشات CRT

تحدثنا في القسمين السابقين عن طرق إدخال الصور رقمياً من خلال الماسحات الضوئية والكاميرات الرقمية، وسنتحدث في هذا القسم عن طرق إخراج الصورة لمشاهدتها، وكيفية تحويلها من ذلك السيل من الأرقام الذي تعرفنا عليه إلى صورة ملونة مرة أخرى.

لعل أشهر الطرق لإخراج الصورة هي شاشات العرض Monitors المصاحبة لجميع أجهزة الحاسوب. تستخدم معظم شاشات العرض طريقة CRT Cathode Ray) (Tube لعرض الصور، وهي ذاتها الطريقة المستخدمة في أجهزة التلفزة. وسنعرض في

هذا القسم لكيفية تحويل الصورة مرة أخرى من سلسلة من الأرقام الممثلة للون إلى شاشة مرئية.

عندما يرسل الحاسوب معلومات الصورة إلى ما اصطلح على تسميته بكرت الشاشة فإن محولاً يقوم بتحويل هذه المجال الخطي إلى مجال تناظري ليتم عرضه على الشاشة. ولعل أشهر هذه المحولات هذه الأيام هو محول SVGA Super Video Graphics) (Array الذي يحتوي بدوره (DAC Digital to Analog Converter) للتحويل من المجال الخطي إلى المجال التناظري. ويجدر بالذكر أن التحويل يتم للألوان الثلاثة: الأحمر، الأخضر والأزرق، كلاً على حدة. حيث يقوم DAC بتحويل قيمة اللون إلى فولطية معينة. ثم يرسلها إلى CRT.



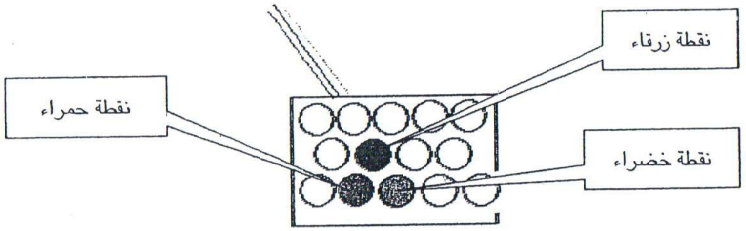
شكل 3. 10 CRT

يحتوي CRT كما في الشكل 10.3 ثلاثة أقسام رئيسة: قاذفة الإلكترونات Electron Gun، التي تصدر حزمة ضيقة من الإلكترونات لكل لون من الألوان الثلاث الرئيسة: الأحمر، الأخضر والأزرق. القسم الثاني هو القطب الموجب Anode وهو المسؤول عن زيادة سرعة حزمة الإلكترونات الصادرة من القاذفة، ذلك أن القطب الموجب يجذب الإلكترونات إليه مما يزيد من سرعتها، كما تعلمنا في الفيزياء من تجاذب

الشحنات المتخالفة. القسم الثالث من أقسام CRT هو ملفان أحدهما أفقي والآخر عمودي وهما مسؤولان عن تغيير مسار شحنة الإلكترونات حتى تسقط على نقطة معينة على الشاشة الفسفورية المضيئة داخل الشاشة. تضيء النقطة على الشاشة الفسفورية عند سقوط حزمة الإلكترونات عليها مما يؤدي إلى انبعاث أشعة لونية هي ما يمثل الصورة في العين. وهناك نقاط فسفورية مخصصة لكل لون من الألوان الثلاثة الرئيسة: الأحمر، الأخضر والأزرق.

عندما ترسم الصورة على الشاشة فإن CRT يبدأ بإسقاط الحزم الإلكترونية على النقاط المضيئة في الشاشة بشكل متسلسل من اليسار إلى اليمين سطراً سطراً حتى تتم إضاءة جميع النقاط على الشاشة. وعندما تنتهي النقاط المضيئة على الشاشة يعود مؤشر CRT من جديد إلى الزاوية العلوية اليسرى من الشاشة ليضيء النقاط الفسفورية من جديد. ويطلق على عدد المرات التي يمكن لجهاز CRT رسم الشاشة المضيئة فيها كل ثانية بمعدل التجديدRefresh Rate، ويساوي في العادة 60 مرة في الثانية.

وهكذا فإن شاشات العرض تقوم بتحويل الأرقام إلى شحنات كهربية مرة أخرى ويتولى جهاز CRT توجيه هذه الشحنة الكهربية إلى نقطة معينة من الشاشة حسب موقع هذه البيكسل. وتضاء النقاط بالألوان الثلاث: الأحمر والأخضر والأزرق إلا أن عين الإنسان قادرة على دمج الألوان المتقاربة مرة أخرى لتكوين الصورة على الشبكية كما في الشكل 11.3.



شكل 3. 11 إعادة تشكيل اللون من النقاط الحمراء والخضراء والزرقاء المتقاربة في CRT

هذا وتختلف شاشة الحاسوب عن تلك المستخدمة في التلفاز، حيث يستلزم الحاسوب تمايزاً أعلى لأننا نجلس أقرب إلى شاشة الحاسوب منا إلى شاشة التلفاز. ويجدر الذكر هنا أن شاشات الحاسوب تختلف في تمايزها وحجمها، حيث يقاس حجم شاشة الحاسوب بالطول القطري. ومن الأمثلة على أحجام الشاشات الشائعة في الحاسوب الشخصي: 14 إنشاً، 15إنشاً و 17 إنشاً. ولا يمثل هذا الطول القطري المساحة المستغلة في الشاشة، أي النقاط المضيئة ولكنه يمثل الطول القطري الكلي للشاشة بما فيها المنطقة البلاستيكية المحيطة بالشاشة الفسفورية، مما يجعل هذا المقياس غير دقيق لكفاءة الشاشة. كما تحتسب القدرة العليا لتمايز الشاشة بمقياس dot pitch الذي يمثل المسافة بالمليمترات بين النقاط الفسفورية ذات اللون الواحد (الأحمر/الأخضر/الأزرق).

وقبيل الانتهاء من هذا القسم الذي خصص لشاشات العرض التي تستخدم تكنولوجيا CRT يجدر بنا أن نذكر أن تكنولوجيا جديدة أصبحت تستخدم حالياً في الشاشات وهي المسماة (LCD Liquid Crystal Display) أي عرض السائل الكريستالي. وتمتاز شاشات LCD بأنها أقل سماكة من تلك التي تستخدم تكنولوجيا CRT سالفة الذكر، كما انها لا تنتج أشعة كهرومغناطيسية كما هي الحال في شاشات CRT. وتبقى شاشات LCD أكثر تكلفة وأقل سرعة في الاستجابة ودقة في اللون.

4. معالجة الصور

بعد التعرف على كيفية تخزين الصورة وتحويلها إلى سلسلة من أرقام الواحد والصفر، فإنه يمكنك أن تتخيل كمية العمليات التي يمكن إجراؤها على هذه الأرقام مما قد يغير الصورة. وإذا كانت هذه العمليات مدروسة وتتبع خوارزميات معدة سابقا، فإنه يمكننا أن نحدث تغييرات ذات أثر إيجابي مدروس على الصورة، كتحسين نوعيتها.

يعرض هذا القسم لبعض الأمثلة العملية على خوارزميات محددة تؤثر على البيانات في الصورة فتغيرها وتغير بالتالي الصورة الناتجة. تخدم هذه الخوارزميات أهدافاً متعددة كتعديل الصورة، أو اقتباس أجزاء منها.

1.4. طرق تنقية الصورة Filters

سبق وأن ذكرنا أن أحد أشهر الاستخدامات لوسائل معالجة الصور هي تلك المختصة بتعديل أخطاء الصورة أو إزالة الشوائب العالقة فيها. حيث تستخدم طرق تنقية الصورة لإزالة مثل هذه الشوائب، وسنتعرف هنا على ثلاث طرق للتنقية: التنقية باستخدام الوسط، التنقية باستخدام الوسيط، والتنقية باستخدام نظام جاوس.

تعرّف التنقية بأنها: "تقليل كمية التغير في اللون/شدة اللون بين البيكسل والبيكسلات المحيطة بها". وتستخدم التنقية عادة لإزالة آثار الشوائب في الصورة.

1.1.4.التنقية باستخدام الوسط Mean Filter

تستخدم طريقة التنقية باستخدام الوسط الحسابي لإزالة الشوائب ذات الحجم الكبير في الصورة، حيث تقوم هذه الطريقة على تعديل قيمة اللون في كل بيكسل بحساب الوسط الحسابي لهذه البيكسل مضافاً إليها البيكسلات المحيطة بها على شكل مصفوفة، كما في المثال التالي:

206	125	98
216	125	98
212	214	101

حيث تحتسب قيمة البيكسل المظللة في المثال كما يلي:

$$155 = \frac{206+125+98+216+125+98+212+214+101}{9} =$$

ويجدر بالذكر أن حجم المصفوفة لا يقتصر فقط على 3 × 3 حيث يمكن استخدام مصفوفة 5 × 5 للحصول على قيمة أكثر وسطية للبيكسل.

ويسهل تمييز مضار هذه الطريقة في التنقية، حيث يتم استبدال قيمة جميع البيكسلات في الصورة بالوسط الحسابي المحيط، إذ تؤثر القيمة الشاذة سلبياً على قيم

البيكسلات المحيطة بها. والمشكلة الثانية التي تواجه تطبيق طريقة التنقية باستخدام الوسط أنها تجعل الصورة أقل وضوحاً خصوصاً عند الحواف بين الأشكال داخل الصورة. حيث تستبدل حواف الأشكال في الصورة بالوسط الحسابي للمحيط، مما يقلل حدة هذه الحواف ووضوحها.

هناك نسخة محسنة من طريقة التنقية باستخدام الوسط وتسمى "الوسط باستخدام الحد" Thresholding Average، حيث يتم تحديد قيمة معينة x ومن ثم يتم حساب الوسط لكل بيكسل. يتم استبدال البيكسل بالوسط الحسابي لها فقط عندما يكون الفرق بين القيمة الأصلية والوسط أعلى من قيمة الحد المعينة x. فوائد هذه النسخة المحسنة أنها لا تغير كل قيمة في الصورة وإنما ما يمكن تصنيفه بالقيم الشاذة أو قيم الشوائب التي يؤدي تغييرها إلى تقليل الفروقات بين البيكسل ومحيطها، مما يؤدي إلى تقليل الشوائب مع أثر أقل على فقدان تفاصيل الصورة.

ويمكن أن تحتسب طريقة التنقية باستخدام الوسط الحسابي باستخدام ضرب المصفوفات Convolution، حيث يتم ضرب المصفوفة 3 × 3 ذات قيم البيكسلات في الصورة بمصفوفة صغيرة مماثلة تسمى kernel لإحداث التأثير. وتكون مصفوفة kernel لطريقة التنقية باستخدام الوسط الحسابي كما يلي:

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$$

فعند ضرب هذه المصفوفة بالمصفوفة في الشكل السابق فإنه تنتج نفس النتيجة سابقة الحساب كما يلي:

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} \bullet \begin{bmatrix} 206 & 125 & 98 \\ 216 & 125 & 98 \\ 212 & 214 & 101 \end{bmatrix}$$

$$155 = 101\times\frac{1}{9}+98\times\frac{1}{9}+98\times\frac{1}{9}+214\times\frac{1}{9}+125\times\frac{1}{9}+125\times\frac{1}{9}+212\times\frac{1}{9}+216\times\frac{1}{9}+206\times\frac{1}{9}=$$

مثال

احسب القيمة الناتجة عن التنقية باستخدام الوسط للبيكسل المظللة في الشكل:

123	212	124
125	255	128
127	127	127

الحل:

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} \bullet \begin{bmatrix} 123 & 212 & 124 \\ 125 & 255 & 128 \\ 127 & 127 & 127 \end{bmatrix}$$

$$150 = 127\times\frac{1}{9}+127\times\frac{1}{9}+127\times\frac{1}{9}+125\times\frac{1}{9}+255\times\frac{1}{9}+128\times\frac{1}{9}+123\times\frac{1}{9}+212\times\frac{1}{9}+124\times\frac{1}{9}=$$

إذن يتم استبدال قيمة البيكسل المظللة بالقيمة 150.

2.1.4. التنقية باستخدام الوسيط Median Filter

الطريقة الثانية للتنقية التي سنتعرف عليها هي باستخدام الوسيط الحسابي Median كبديل للوسط الحسابي. وتستخدم هذه الطريقة في إزالة الشوائب النقطية، أي

ذات المساحة الضئيلة بحيث تشمل بيكسلا أو اثنين، ويطلق على هذا النوع من الشوائب اسم Salt and Pepper أي الملح والبهار، حيث تنتشر هذه الشوائب على سطح الصورة كنقاط كثيرة وصغيرة. وتقوم طريقة التنقية باستخدام الوسيط باستبدال قيمة الصورة بالوسيط الحسابي للنقاط الموجودة في المصفوفة 3 × 3 المحيطة بالصورة مثلا، كما في:

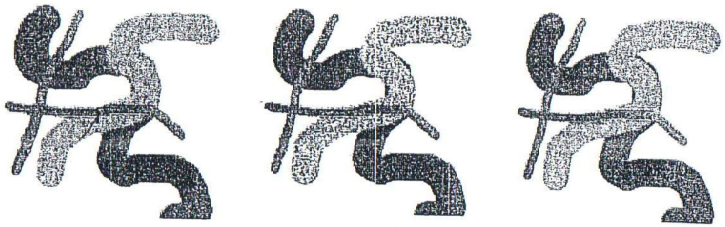
206	204	200
198	25	205
212	211	207

حيث يحتسب الوسط الحسابي من خلال إعادة ترتيب القيم في المصفوفة ترتيبا تصاعديا: 25، 198، 200، 204، 205، 206، 207، 211، 212. ثم تحتسب منزلة الوسيط كما يلي: (9+1)/2=5 حيث يمثل العدد 9 عدد الأرقام المرغوب حسابة الوسيط لها. ويحتسب الرقم ذو المنزلة 5 كوسيط للمجموعة، وهو في هذه الحالة 205. عند النظر إلى المصفوفة الناتجة:

206	204	200
198	205	205
212	211	207

فإنه يسهل ملاحظة أن البيكسل الذي كان يمثل قيمة شاذة (أقل بكثير من بقية القيم) تم التخلص من قيمته باحتساب الوسيط. وكلما كانت الشوائب ذات حجم أكبر، احتجنا إلى مصفوفات أكبر لاحتساب الوسيط: 5 × 5، 7 × 7 وهكذا.

وبالرغم من نجاعة طريقة التنقية باستخدام الوسيط الحسابي في التخلص من الشوائب صغيرة الحجم، إلا أنها تؤدي بالطبيعة إلى تقليل حدة التغاير أو التباين في الصورة الناتجة كما في الشكل 12.3.



الصورة بعد التنقية بالوسيط | الصورة + الشوائب | الصورة الأصلية

شكل 12.3 التنقية باستخدام الوسيط Median

تختلف التنقية باستخدام الوسيط عنها باستخدام الوسط الحسابي في أنها تقلل أثر القيمة الشاذة على محيطها، إلا أنها تحتاج إلى وقت أطول في الحساب. وذلك لأننا نحتاج إلى قراءة القيم في كل مصفوفة، ومن ثم إعادة ترتيبها، وتحديد الوسيط، قبل تحديد القيمة الجديدة للبيكسل.

نشاط

يمكنك تطبيق الفكرة الواردة في المثال السابق والتنقية باستخدام الوسيط باستخدام معالج الصور Photoshop. قم بإضافة شوائب إلى الصورة باستخدام الأمر Filter -> Noise -> Add Noise.

استخدم الأمر: Filter -> Noise -> Median للتنقية باستخدام الوسيط.

قم بإعادة الأمر السابق عدة مرات للحصول على نتيجة أفضل.

3.1.4. التنقية بطريقة Gaussian

في التنقية بطريقة جاوس Gauss، فإن هذه الطريقة تعطي وزناً أكبر في الحساب للبيكسلات القريبة من المركز في المصفوفة. أي أن أثر قيمة البيكسل يتناقص مع بعدها عن المركز. ومن الأمثلة على التنقية باستخدام طريقة جاوس المصفوفة التالية:

$$\begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix}$$

تبين المصفوفة السابقة كيف يعطى وزن أكبر في حساب القيمة للبيكسل الأصلية والبيكسلات الأربع المحيطة بالجهات الأربع. كما في المثال التالي:

$$\begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix} \bullet \begin{bmatrix} 206 & 125 & 98 \\ 216 & 125 & 98 \\ 212 & 214 & 101 \end{bmatrix}$$

$$= 212\times\frac{1}{16}+214\times\frac{2}{16}+101\times\frac{1}{16}+216\times\frac{2}{16}+125\times\frac{4}{16}+98\times\frac{2}{16}+206\times\frac{1}{16}+125\times\frac{2}{16}+98\times\frac{1}{16}=151$$

تستخدم التنقية بطريقة جاوس للتخلص من الشوائب غير المنتظمة في الصورة ولإعطاء قدر أكبر للقيمة الأصلية للبيكسل في الحساب.

مثال:

38	44	38
35	216	26
23	33	33

قم بتنقية القيمة المظللة للبيكسل باستخدام:

1. الوسط الحسابي
2. الوسيط الحسابي
3. طريقة Gauss

أي الطرق الثلاثة السابقة أنسب في التخلص من هذه الشائبة؟ لماذا؟

الحل:

1. باستخدام الوسط الحسابي

$$\text{القيمة الجديدة} = \frac{38+44+38+26+216+35+33+33+23}{9} = 54$$

2. باستخدام الوسيط الحسابي

القيم مرتبة هي: 23، 26، 33، 33، 35، 38، 38، 44، 216

القيمة الجديدة هي 35

3. باستخدم مصفوفة Gauss

القيمة الجديدة =

$$23\times\frac{1}{16}+33\times\frac{2}{16}+33\times\frac{1}{16}+35\times\frac{2}{16}+216\times\frac{4}{16}+26\times\frac{2}{16}+38\times\frac{1}{16}+44\times\frac{2}{16}+38\times\frac{1}{16}$$

$$= 80$$

الطريقة الأنسب في التخلص من هذه الشائبة هي الوسيط الحسابي لأنها لا تحتسب أية قيمة للبيكسل الشاذة. أما الوسط الحسابي فهي تحتسب القيمة الشاذة والقيم المتجاورة بنفس المقدار. في حين تعتبر مصفوفة Gauss أقل الطرق أثراً في التخلص من اثر هذه الشائبة لأنها تعطي القيمة الأكبر في الحساب للبيكسل المتوسطة.

2.4. التعديل

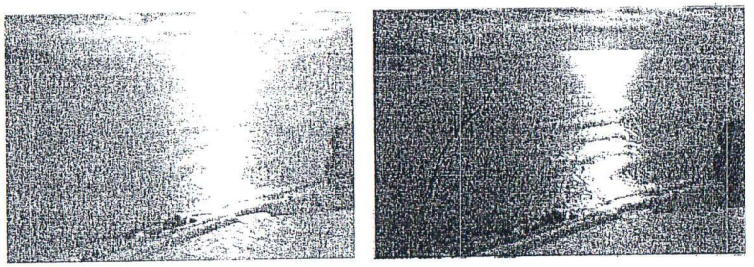
استعرض القسم السابق لإزالة الشوائب باستخدام طرق التنقية المختلفة، إلا أن هذا الاستخدام ليس الوحيد لطرق معالجة الصور كما سبق وذكرنا. يعرض هذا القسم طرق تعديل إضاءة الصورة.

كثيراً ما التقطنا صوراً فوتوغرافية شخصية في أماكن مضيئة فظهر الوجه أكثر

سواداً مما يتوجب، أو التقطنا صوراً في إضاءة ضعيفة فغابت ملامحنا فيها. ولأن الصور الفوتوغرافية على اختلاف أزمانها وأمكنتها تحمل معها ذكريات جميلة، فإنه يمكن تعديل مثل هذه الصور باستخدام طرق التعديل المختلفة للصورة. يستعرض هذا القسم طريقتين رئيستين في تعديل إضاءة الصورة: التعديل الخطي، والتعديل اللوغاريتمي.

1.2.4. التعديل الخطي Linear Mapping

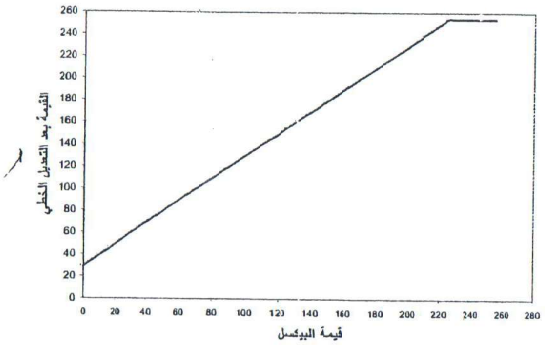
طريقة التعديل الخطي تهدف أساساً إلى زيادة الإضاءة في الصورة أو تقليلها. ففي الشكل 13.3 مثلاً، نلاحظ أن الإضاءة ليست بالقدر الكافي، في حين أن الصورة المعدلة يظهر فيها البحر بشكل أفضل.



الصورة المعدلة تعديل خطياً　　الصورة الأصلية

شكل 3. 13 التعديل الخطي
الصورة: شرم الشيخ – مصر (2005)

يقوم التعديل الخطي، ببساطة، بزيادة قيمة كل بيكسل بمقدار ثابت. مما يؤدي إلى اقتراب قيمة البيكسل من اللون الأبيض، وبالتالي تفتيح الصورة. فمثلاً لو كانت الزيادة بمقدار 30، وكانت قيمة بيكسل معينة في الصورة تساوي 112 فإن القيمة الجديدة لهذه البيكسل = 112+30 = 142، وهكذا بالنسبة لكل بيكسل في الصورة. ويوضح الشكل 14.3 الاقتران الممثل لهذه العلاقة:

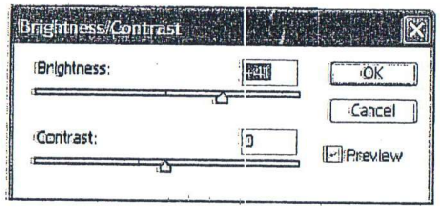


شكل 3. 14 الاقتران الممثل لعلاقة التغير الخطي

ويجدر بنا هنا أن نذكر أن البيكسلات البيضاء أصلاً لا تغير قيمتها عند التعديل الخطي، بل تحافظ عليها. ذلك لأننا تعلمنا أن قيمة البيكسل لا يمكن أن تزيد عن 255 وهي القيمة القصوى في التدرج الرمادي. فإذا كانت البيكسل تحمل قيمة 240 في البداية فإن قيمتها بعد التعديل بمقدار 30 تساوي 255، حيث لا يمكن لأي قيمة أن تتجاوز هذا الحد الأعلى.

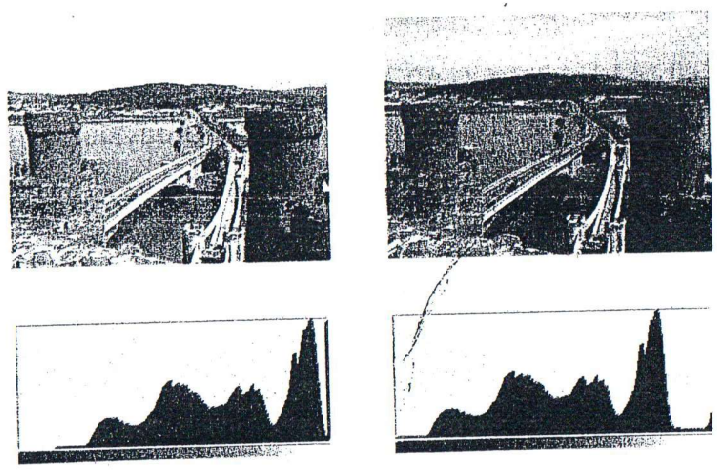
نشاط

لإجراء تعديل خطي باستخدام برنامج تعديل الصور Photoshop، إفتح الصورة المراد تعديلها باستخدام البرنامج ثم استخدم الأمر Image -> Adjustments -> Brightness/Contrast...



عن طريق زيادة أو تقليل Brightness في الصورة يمكنك إجراء تعديل خطي.

ولفهم هذه الطريقة بشكل أفضل، نود هنا أن نعرّفك على مصطلح Histogram أو التمثيل البياني لقيم الصورة، ويرسم بالطريقة التالية: يتكون رسم Histogram من محوري السينات والصادات. يمثل محور السينات القيم التي تأخذها البيكسل من 0 إلى 255 في حين يمثل محور الصادات تكرار هذه القيمة في الصورة، أي عدد البيكسلات التي تحمل هذه القيمة في الصورة. وسنعرض أولاً للتمثيل البياني في الصور ذات التدرج الرمادي Gray Scale ويظهر الشكل 15.3 التمثيل البياني لصورتين بالتدرج الرمادي قبل وبعد التعديل الخطي:

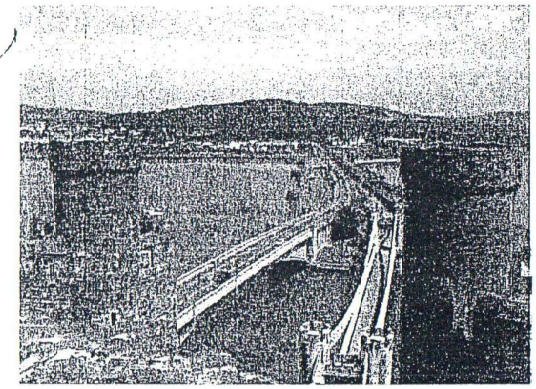


الصورة المعدلة تعديل خطياً　　الصورة الأصلية

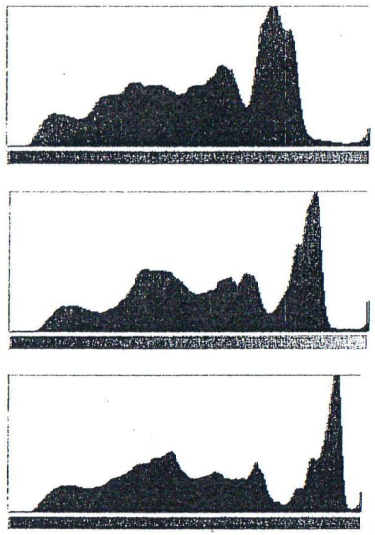
شكل 3 15 تغير Histogram اثناء التعديل الخطي
الصورة: Wales Castle – ويلز – المملكة المتحدة (2003)

ويظهر من التمثيل البياني للصورتين أن كل بيكسل تحركت مقداراً ثابتا إلى اليمين أي باتجاه اللون الأبيض. كما يبين التمثيل البياني للصورة المعدّلة أن البيكسلات ذات القيمة العالية تراكمت عند القيمة 255 وهي قيمة اللون الأبيض الناصع حيث لا يمكن تجاوزها كما سبق وشرحنا.

أما فيما يتعلق بالصور الملونة، فإن هناك تمثيلاً بيانياً Histogram لكل من أقسام الفضاء اللوني Channels. ففي الصورة التالية ذات الفضاء اللوني RGB:



فإن هناك تمثيلاً بيانياً لكل من الألوان الثلاثة الرئيسة: الأحمر والأخضر والأزرق، كما يلي:

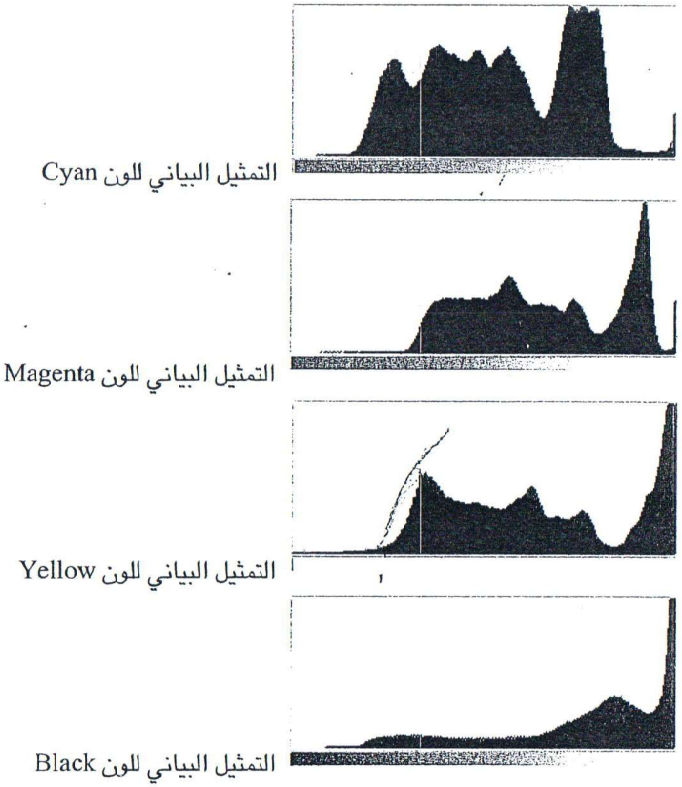


التمثيل البياني للون الأحمر

التمثيل البياني للون الأخضر

التمثيل البياني للون الأزرق

أما إذا قمنا بتحويل هذه الصورة إلى الفضاء اللوني CMYK باستخدام الأمر Image -> Mode فإننا نحصل على التمثيل البياني للألوان الأربعة: Cyan، Magenta، Yellow وBlack كما يلي:

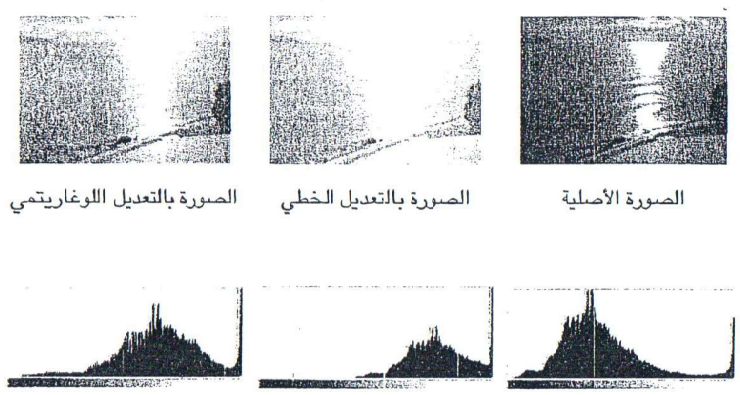


2.2.4.التعديل اللوغاريتمي Logarithmic Mapping

برغم الفوائد الجمة التي يقدمها التعديل الخطي للصور إلا أن هذا التعديل يرفع قيمة كل بيكسل من الصورة بمقدار ثابت، ولا تنطبق هذه الحالة عندما يكون الجسم المراد

تحسين صورته داكناً على خلفية بيضاء، حيث تزداد المساحة البيضاء في الصورة بدل تعديلها.

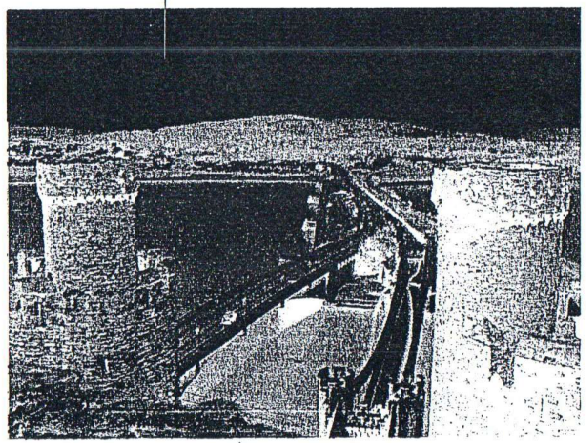
يقوم التعديل اللوغاريتمي على أساس زيادة الفروق اللونية بين البيكسلات الداكنة بشكل أكبر من تلك الفروقات بين البيكسلات الفاتحة، فالقيم الداكنة (الصغيرة) تتباعد، في حين تتقارب القيم الفاتحة. ويمكن لهذا النوع من التعديل إحداث التغيير المطلوب في توضيح أجسام داكنة قليلاً في الصورة دون تفتيح الخلفية.



شكل 3. 16 الفرق بين التعديل الخطي والتعديل اللوغاريتمي على الصورة والHistogram

يعرض الشكل 16.3 الفرق بين التعديل الخطي والتعديل اللوغاريتمي. ففي حين أن التعديل الخطي أدى إلى تفتيح جميع الألوان في الصورة، أدى التعديل اللوغاريتمي إلى تفتيح البحر بشكل أفضل وإبقاء المناطق الداكنة (كالصخرة والجبل) داكنة بشكل يوضح الفروق في الصورة. كما يسهل أن نلحظ أثر التعديل اللوغاريتمي على التمثيل البياني لقيم البيكسلات Histogram في المثال السابق. حيث أدى التعديل اللوغاريتمي إلى توزيع أكثر انتظاماً للبيكسلات على التدرج.

3.4. قلب الصورة Invert



شكل 3. 17 قلب الصورة Invert

انظر إلى الصورة في الشكل 17.3، لقد تعارفنا على هذه الصورة بأنها negative (نيجاتيف الصورة). إن النيجاتيف ينتج عن قلب كل قيمة في الصورة إلى القيمة المعاكسة. في التدرج الرمادي فإنه يمكن احتساب قيمة كل بيكسل في الصورة بعد القلب حسب المعادلة:

القيمة الجديدة = 255 – القيمة الأصلية

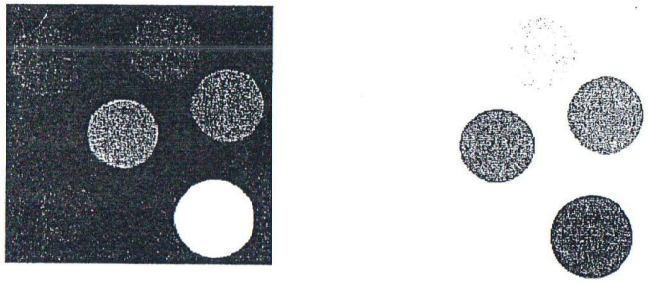
فإذا كانت البيكسل بيضاء (أي 255) قبل القلب فإنها تصبح سوداء تماماً (أي 0) بعد القلب.

أما في الصور الملونة، فإن القلب يتم لكل قيمة في الفضاء اللوني Channel على حدة، ففي الفضاء اللوني RGB تكون القيم بعد القلب كما يلي:

قيمة اللون الأحمر الجديدة = 255 – قيمة اللون الأحمر القديمة

قيمة اللون الاخضر الجديدة = 255 – قيمة اللون الأخضر القديمة

قيمة اللون الأزرق الجديدة = 255 – قيمة اللون الأزرق القديمة

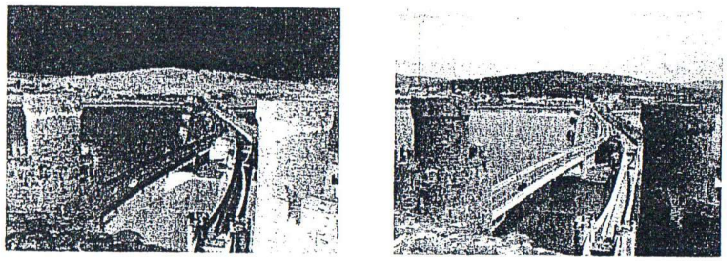


الصورة بعد القلب　　الصورة الأصلية

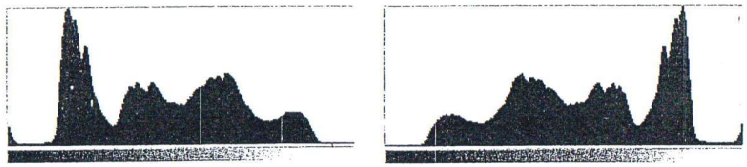
شكل 3. 18 مثال 2 لقلب الصورة Invert

لاحظ في الشكل 18.3 كيف تم استبدال اللون الأزرق مثلاً (0، 0، 255) باللون الأصفر (255، 255، 0)... واللون (Magenta 0، 255، 255) باللون الأخضر (0، 255، 0) وهكذا.

يمكنك أن تستنتج أن التمثيل اللوني للصورة قبل وبعد القلب يقوم على قلب التمثيل البياني أفقياً Flip Horizontally كما في الشكل 19.3:



الصورة بعد القلب　　الصورة الأصلية



التمثيل البياني للصورة الأصلية | التمثيل البياني للصورة بعد القلب

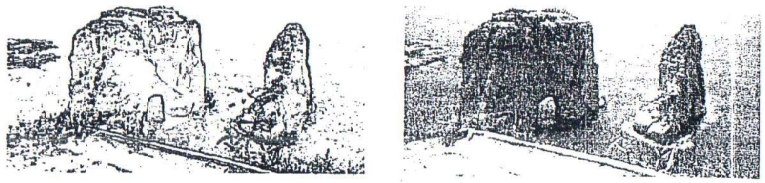
شكل 3. 19 مواقع الألوان

4.4. تحديد الحواف Edge Detection

تعرف الحواف في الصورة بأنها المناطق التي يحدث فيها تغيير مفاجئ في قيمة البيكسل، أي تغيير حاد في شدة اللون أو درجته. ويمكن الاستفادة من هذه الحواف في تحديد الأشكال الموجودة في الصورة لتعريفها لاحقاً والتعامل معها. ويعد موضوع تحديد الحواف موضوعاً مؤسساً لعلم "بصر الحاسوب" أو "Computer Vision". حيث يمكن التعرف على الأشكال في الصورة ودراستها بعد تحديد الحواف. فمثلاً يمكننا دراسة الأشكال المختلفة في الصورة بُعَيد تحديد الحواف لتحديد الأشكال الدائرية أو المستطيلة.

إلا أن عملية تحديد الحواف ليست بالسهولة التي عرضنا لها. ذلك أن الشوائب، إثر الإضاءة وغيرها من الخواص غير المستحبة لها تأثير سلبي على دقة تحديد الحواف. لذا يجدر استخدام طرق التنقية في إزالة الشوائب قبيل تحديد الحواف. ونود أن نستذكر هنا ما شرحناه في الفصل 2.5. حول طرق التنقية، حيث وضحنا أن طرق تنقية الشوائب المختلفة تؤدي إلى تقليل حدة الصورة وتباينها، مما يؤثر سلباً على حدة الحواف وبالتالي استخلاصها. لذا فإن تحديد الحواف لا يمكن أن يكون مثالياً أو كاملاً.

وتوضح الصورة 20.3 الحواف الناتجة عن عملية "تحديد الحواف" في الصورة.



الصورة الأصلية | حواف الصورة

شكل 3. 20 تحديد الحواف

الصورة: صخرة الروشة – بيروت – لبنان (2005)

إحدى الخوارزميات المبسطة المستخدمة لحساب الحواف هي تلك التي قام بإعدادها العالم Sobel وتسمى بتقنية سوبل لتحديد الحواف. حيث تقوم على حساب الفرق طولياً وعرضياً بين البيكسل والبيكسلات المجاورة لها لتحديد إن كانت هذه البيكسل حداً/حافة طولية أو عرضية.

وتستخدم المصفوفتان التاليتان في الحساب كما يلي:

$$\text{هـ(س)} = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad \text{هـ(ص)} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

حيث تستخدم المصفوفة الأولى لحساب الحدود الأفقية في الصورة وتستخدم المصفوفة الثانية لحساب الحدود العمودية في الصورة كما في المثال التالي:

في المصفوفة 3 × 3 التي تحتوي قيم البيكسل فإننا نقوم بضرب المصفوفات كما شرحنا في القسم 2.5.

$$\text{م} = \begin{bmatrix} 100 & 100 & 100 \\ 100 & 100 & 100 \\ 20 & 20 & 20 \end{bmatrix}$$

في البيكسلات المتجاورة ذات القيم المتتالية فإنه يسهل أن نلحظ أن البيكسل ذات الخلفية الرمادية تشكل حداً عمودياً لا أفقيا ذلك أنها تختلف وبشكل ملحوظ في القيمة عن البيكسل التي تقع تحتها مباشرة. وبحساب قيم الحدود باستخدام مصفوفتي سوبل فإن القيم تكون كالآتي:

$$\text{هـ(س)} \times \text{م} = +100\times1 - +100\times0 + 100\times1 + 100\times2 + 100\times0 + 100\times2 - +$$

$$20\times1 + 20\times0 + 20\times1 - = 0$$

$$\text{هـ(ص)} \times \text{م} = +100\times1 - +100\times2 - +100\times1 - +100\times0 + 100\times0 + 100\times0 +$$

$$20\times1 + 20\times2 + 20\times1 = 320 -$$

ويوضح المثال كيف كانت قيمة الحد الأفقي صفراً (لا تغيير) في حين حمل الحد/الحافة العمودية قيمة سالبة عالية مما يشير إلى وجود حد عمودي عند هذه البيكسل وتمثل البيكسل بالتالي حداً بين منطقتين.

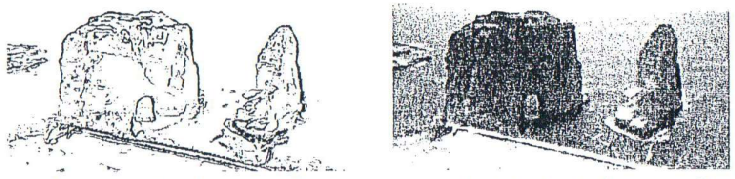
ومن السهل أن نلاحظ أن قيمة الحد تكون أكبر كلما كان التغير أكبر وأوضح، وبالتالي يمكننا الاعتماد على القيم الكبيرة في نتيجة المعادلات أعلاه في تحديد الحواف واستثناء القيم ذات التغير الصغير التي يمكن أن تكون قد نجمت عن الشوائب أو تغير طفيف في الإضاءة.

وبنهاية حديثنا عن الحواف، نكون قد أنهينا القسم الخامس من هذه الوحدة والذي تناول معالجة الصور والطرق الأساسية في معالجة الصور.

نشاط

باستخدام معالج الصور Photoshop، يمكنك استخراج حواف الصورة باستخدام الأمر: Filter->Stylize-> Find Edges. استخدم الأمر سالف الذكر في استخراج حواف مجموعة من الصور المخزنة لديك.

سبق وشرحنا في القيم 2.1.5. أن لطرق التنقية أثراً سلبياً على وضوح الحواف في الصورة. يمثل الشكل التالي أثر التنقية باستخدام الوسيط على حواف الصورة كما يبين الشكل 21.3. يمكنك إجراء هذا النشاط أيضاً باستخدام برمجية Adobe Photoshop.

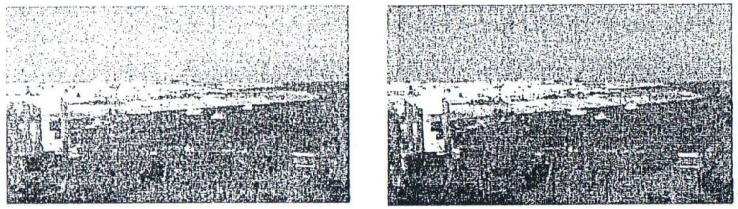


الصورة بعد التنقية باستخدام الوسيط — حواف الصورة بعد التنقية باستخدام الوسيط

شكل 3. 21 أثر التنقية باستخدام الوسيط على حواف الصورة

5.4. التغيير على إحدى قنوات الفضاء اللوني (RGB Channels)

يمكن إجراء بعض التعديلات على قناة واحدة Channel في الصور الملونة بدل إحداث التغيير في القنوات الثلاث معاً كما في المثال التالي:



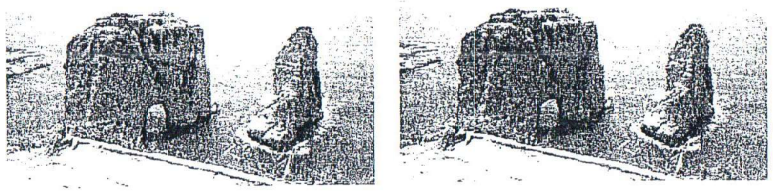
الصورة الأصلية — التعديل الخطي بزيادة قيم اللونين الأحمر والأخضر فقط

شكل 3. 22 التعديل الخطي على بعض قنوات الفضاء اللوني RGB

في الشكل 22.3، تم تعديل اللونين الأحمر R والأخضر G فقط في الصورة، ولفهم

ذلك بشكل أفضل لاحظ كيفية تحول الألوان إلى اللون الأصفر (مزيج الأحمر والأخضر).

يمكن تعديل قناة واحدة فقط في الصورة فقط كما يلي:



الصورة الأصلية

التعديل الخطي بتقليل قيم اللون الأحمر

شكل 3. 23 التعديل الخطي على قناة اللون الأحمر في **RGB** فقط

لاحظ في الصورة إلى يسار الشكل23.3 كيف أصبح البحر مائلاً إلى اللون Cyan وهو خليط الأزرق والأخضر مما يشير إلى نقصان قيمة اللون الأحمر.

نشاط

يمكن تعديل قيم بعض قناة دون غيرها باستخدام Adobe Photoshop باستخدام الأمر: Image -> Adjustments -> Color Balance

5. نموذج ملف Bitmap

يعد ملف Bitmap النوع الأساسي لملفات الصور على نظام التشغيل ويندوز، وقد تم تصميم هذا التكوين لملف Bitmap من قبل شركة مايكروسوفت أيضاً بهدف تخزين الصور. وتنتهي ملفات Bitmap بـ .bmp أو .dip – اختصارا ل Device-Independent Bitmap، ويعني أنها تعرض بنفس الطريقة بغض النظر عن الوسيلة التي تعرض عليها هذه الصورة- . وعادة ما تكون ملفات Bitmap غير مضغوطة إلا

أنه يمكن ضغط هذه الملفات باستخدام الترميز الكلي Run-length Encoding (راجع القسم 1.2. من الوحدة الخامسة)

يقسم ملف Bitmap إلى أربعة أقسام رئيسية:

1. ترويسة الملف File header وتتكون من bytes 14
2. ترويسة معلومات (Bitmap Information Header) وتتكون من bytes 40
3. جدول الألوان Color Table وتحتوي معلومات عن الألوان الموجودة.
4. معلومات البيكسلات Pixel Data ويحتوي المعلومات اللونية لكل بيكسل في الملف.

وهي المعلومات التي كنا نتحدث عنها في الأقسام السابقة.

ترويسة الملف	ترويسة المعلومات	جدول الألوان	معلومات البيكسلات
bytes 14	bytes 40		

وفيما يلي شرح كلّ من هذه الأقسام :

ترويسة الملف **File Header**:

تحتوي ترويسة الملف معلومات حول الملف، وتقسم حسب الجدول 5.3:

نوع الملف	bytes 2	وتساوي دائما BM أي ملف Bitmap
حجم الملف	bytes 4	وهو حجم الملف كاملاً بالبايت
محجوز	bytes 4	وهي منطقة محجوزة في الترويسة وتساوي صفراً دائماً
فاصل المسافة Offset	bytes 4	المسافة بين بداية الملف وبداية معلومات البيكسلات

جدول 3. 5 ترويسة الملف **File Header** في ملف **Bitmap**

ترويسة المعلومات **Bitmap Information Header**:

في حين تحتوي ترويسة الملف سالفة الذكر معلومات عامة عن الملف، فإن ترويسة معلومات Bitmap تشتمل على معلومات خاصة بالصورة: إحداثياتها، تمايزها، وألوانها، وتتكون ترويسة الملف من 40 bytes تشتمل على المعلومات كما في الجدول 6.3:

حجم ترويسة المعلومات	4 bytes	عدد bytes المستخدمة لتسجيل ترويسة المعلومات وتساوي دائماً 40 bytes.
عرض الصورة	4 bytes	وهو عرض الصورة بالبيكسلات
ارتفاع الصورة	4 bytes	وهو ارتفاع الصورة (أو طولها) بالبيكسلات
عدد سطوح الصورة	2 bytes	وتساوي عادة 1
التمايز اللوني	2 bytes	وهو عدد bits المستخدمة لتمثيل اللون في كل بيكسل (راجع القسم 2.2.)
ضغط الصورة	4 bytes	ويمثل نوع الضغط المستخدم في الصورة. فإذا كان = 0 فإن الصورة غير مضغوطة وإذا كان = 1 فإن الصورة تستخدم الترميز الكلي Run-Length باستخدام 8 bits وإذا كان = 2 فإن الصورة تستخدم الترميز الكلي Run-length باستخدام 4 bits، وهكذا
حجم الصورة	4 bytes	وهو عدد bytes المستخدمة في تمثيل معلومات البيكسلات، وتستخدم في حالات الضغط حيث يختلف عدد bytes عن المتوقع. أما عندما يكون الضغط = 0 فإن هذا القسم من ترويسة المعلومات يساوي صفراً أيضاً.

التمايز الأفقي	4 bytes	ويساوي عدد البيكسلات لكل متر أفقي، ويساوي صفراً عادة حيث يعتمد التمايز على وسيلة العرض.
التمايز الرأسي	4 bytes	ويساوي عدد البيكسلات لكل متر عمودي، ويساوي صفراً حيث يعتمد التمايز عادة على وسيلة العرض.
عدد الألوان	4 bytes	ويمثل العدد المختلف من الألوان المستخدمة في الصورة. وإذا كان عدد الألوان هنا يساوي صفراً فإن عدد الألوان يحتسب من التمايز اللوني المذكور سابقا حيث يساوي عدد الألوان $2^{\text{التمايز اللوني}}$.
عدد الألوان المهمة	4 bytes	ويمثل عدد الألوان المهمة في الصورة، وإذا كان يساوي صفراً فإن كل ألوان الصورة تعطى نفس الأهمية.

جدول 6.3 ترويسة المعلومات **Information Header** في ملف **Bitmap**

وبالتالي فإن ترويسة المعلومات تحوي كل المعلومات التفصيلية عن الصورة وتركيبها.

مثال:

صورة غير مضغوطة ذات تمايز 250 × 250 بيكسلاً وذات تمايز لوني بمقدار 4 bits تمثل ترويسة المعلومات لهذه الصورة كما يلي:

حجم الترويسة	عرض الصورة	ارتفاع الصورة	عدد السطوح	التمايز اللوني	ضغط الصورة	حجم الصورة	التمايز الأفقي	التمايز الرأسي	عدد الألوان	الألوان الهامة
40	250	250	1	4	0	0	0	0	0	0
00101000	11111010	11111010	00000001	00000100	00000000	00000000	00000000	00000000	00000000	00000000

جدول الألوان **Color Table**:

ويمثل جدول الألوان القيم اللونية المستخدمة في الصورة. فمثلاً إذا كان التمايز اللوني للصورة = 1 فإنه يمكن تمثيل لونين في الصورة هما الأبيض والأسود أو أي لونين آخرين. يحوي جدول الألوان قيم الأحمر، الأخضر، والأزرق لكل لون من هذه الألوان، حيث يتخذ كل لون من الألوان سطراً في جدول الألوان كما في الجدول 7.3:

قيمة الأحمر	byte 1	وتمثل مقدار اللون الأحمر في اللون الممثل
قيمة الأخضر	byte 1	وتمثل مقدار اللون الأخضر في اللون الممثل
قيمة الأزرق	byte 1	وتمثل مقدار اللون الأزرق في اللون الممثل
محجوز	byte 1	ويساوي صفراً دائماً

جدول 7. 3 جدول الألوان **Color Table** في ملف **Bitmap**

وبالتالي فإنه في المثال السابق حيث يجب تمثيل لونين فقط هما الأبيض والأسود، فإن التمايز اللوني = 1 bit وبالتالي فإن جدول الألوان يتخذ الشكل التالي:

Bit	قيمة الأحمر	قيمة الأخضر	قيمة الأزرق	محجوز
0	0	0	0	0
1	255	255	255	0

ويمكن لنفس التمايز اللوني (1 bit) أن يمثل لونين آخرين كالأزرق والأصفر مثلا كما في الجدول التالي:

Bit	قيمة الأحمر	قيمة الأخضر	قيمة الأزرق	محجوز
0	0	0	255	0
1	255	255	0	0

حيث يمثل bit=0 اللون الأزرق و bit=1 اللون الأصفر، وبالتالي فإن الجدول اللوني يستخدم في شرح ألوان الصورة. ويجدر بنا أن نذكر أن جدول الألوان يستخدم عندما يقل التمايز اللوني عن 8 bits أما فيما يزيد على ذلك فإن الألوان تصبح معروفة ومحددة (كالتدرج الرمادي في حالة 8 bits، وRGB في حالة 24 bits).

معلومات البيكسلات **Pixel Data**:

ويمثل هذا القسم المعلومات اللونية لكل بيكسل على حدة، سواء كانت عبارة عن مؤشراً لقيم معينة في جدول الألوان أو قيم مباشرة كما في قيماً RGB المباشرة.

إلا أن معلومات البيكسلات تكون ممثلة بعكس ما تعوّدنا حيث يمثل السطر الأسفل في الصورة أولا، وهكذا صعودا حتى السطر الأول في الصورة. أي أن المعلومات تخزن من الأسفل للأعلى ويخزن كل سطر في الصورة من اليسار إلى اليمين.

يعد ملف Bitmap أحد أشهر تكوينات ملفات الصور. ولفهم أي تشكيل لملف آخر مثل ملفات Tiff أو غيرها فإن الأساس لمثل هذه التكوينات متشابه، إلا أن طريقة الترتيب، المناطق المحجوزة وأحجام هذه الأقسام تختلف من نوع لآخر. أردنا في هذا القسم أن نعرض طريقة تمثيل أحد هذه الملفات علها تكون مساعداً لك على فهم تمثيل الصور ومعالجتها.

ويسهل أن تجد كثيراً من الخوارزميات المكتوبة التي تتعامل مع ملفات Bitmap وغيرها، فتقرأ بياناتها وتحللها أو تصنع ملفات Bitmap بنفسها. كما في الخوارزمية التالية باستخدام لغة Java:

```
import java.io.*;

public class ReadBitmap {
  public static void main(String[] args) {
      DataInputStream s = null;
      try{
      s = new DataInputStream (new FileInputStream ("1.bmp"));
         // 1.bmp is the file's name
```

```
    byte b;
    // Find the BM characters
    for (int i = 0; i < 2; i++)
    {  b = s.readByte();
      Character ch = new Character ((char)b);
       System.out.print (ch + " ");
    }
    System.out.println();
    // 4 Bytes representing the size of the file
    System.out.print ("File Size = ");
    long totSize = 0;
    for (int i = 0; i < 4; i++)
    {  b = s.readByte();
       int b2 = (int) b;
       if (b2 < 0)
         b2 = 128+(0-b2);
       int toBeAdded = b2;
       for (int j = 0; j < i; j++)
          toBeAdded*=256;
       totSize += toBeAdded;
       System.out.print (b2 + " ");
    }
    System.out.print (" Equals = " + totSize + " Bytes.");
    System.out.println();
    //4 reserved bytes = 0
    System.out.print ("Reserved = ");
    for (int i = 0; i < 4; i++)
    {  b = s.readByte();
       int b2 = (int) b;
       if (b2 < 0)
         b2 = 128+(0-b2);
       System.out.print (b2 + " ");
    }
    System.out.println();
    //4 offset
    System.out.print ("Offset = " );
    for (int i = 0; i < 4; i++)
    {  b = s.readByte();
       int b2 = (int) b;
       if (b2 < 0)
         b2 = 128+(0-b2);
       System.out.print (b2 + " ");
    }
    System.out.println();
    // You can read on bytes in the same way and understand them
    s.close();
    }catch (IOException e)
    { System.out.println (e.getMessage());};
  }
}
```


لإحداث تعديل خطي مثلاً لقيمة البيكسلات في ملف Bmp فإن الخوارزمية التالية تقوم بالمهمة بكفاءة:

```
import java.io.*;

public class LinearMapping {
  public static void main(String[] args) {
     DataInputStream s = null;
     try{
     s = new DataInputStream (new FileInputStream ("1.bmp"));
     DataOutputStream x = new DataOutputStream (new
FileOutputStream (new File ("2.bmp")));
        // 1.bmp is the file's name
     byte b;

     for (int i = 0; i < 34; i++)
     {  b = s.readByte();
        x.writeByte(b);
     }

     int totSize = 0;
     for (int i = 0; i < 4; i++)
     {  b = s.readByte();
        x.writeByte(b);
        int b2 = (int) b;
        if (b2 < 0)
          b2 = 128+(0-b2*128);
        int toBeAdded = b2;
        for (int j = 0; j < i; j++)
           toBeAdded*=256;
        totSize += toBeAdded;
        System.out.print (b2 + " ");
     }

     for (int i = 0; i < 16; i++)
     {  b = s.readByte();
        x.writeByte(b);
     }

     // Image DATA
     System.out.println ("Start of File Info");
     for (int i = 0; i < totSize; i++)
     {
         b = s.readByte();
         int b2 = (int) b;
        if (b2 < 0)
          b2 = 128+(0-b2*128);
         b2 += 60;
         if (b2 > 255)
```

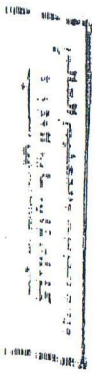
```
            b2 = 255;
          x.writeByte(b2);
      }

      s.close();
      x.close();
      }catch (IOException e)
{ System.out.println (e.getMessage());};


  }
}
```

نشاط

قم باستخدام خوارزمية مماثلة للمثال أعلاه لإجراء تعديل خطي بإنقاص قيمة كل بيكسل بمقدار 40.

نشاط (2)

قم بإنشاء خوارزمية أخرى لإحداث تنقية باستخدام الوسط أو الوسيط الحسابي



- تتكون الصورة من مجموعة نقاط Pixels.
- تتداخل هذه النقاط Pixels في دماغ الإنسان لتشكل الصورة الضوئية.
- يمثل الطرف العلوي الأيسر من الصورة نقطة الأصل في الصورة.
- تمايز الصورة Image Resolution هو عدد البيكسلات الممثلة للصورة.
- التمايز اللوني Color Resolution هو عدد الbits الممثلة للون في كل بيكسل في الصورة.
- عدد الألوان = $2^{\text{التمايز اللوني}}$.
- حجم ملف الصورة = تمايز الصورة × التمايز اللوني

Color Resolution × Image File Size = Image Resolution

- الفضاء اللوني Color Space نظام لوني لفهم علاقة التمايز اللوني بالألوان في الصورة ومن أمثلته: RGB، CMYK، HSI وYUV.
- الفضاء اللوني RGB هو تمايز لوني بمقدار 24 bits مقسمة إلى: Byte1 للون الأحمر، Byte1 للون الأخضر وByte1 للون الأزرق، وقد شاع تمثيله باستخدام النظام السادس عشر.
- يتمثل اللون الأسود في الفضاء اللوني RGB بالقيمة (0، 0، 0) في حين يتمثل اللون الأبيض بالقيمة (255، 255، 255).
- الفضاء اللوني CMY اختصار للألوان الثلاثة: Cyan، Magenta وYellow، وقد استحدث لأهميته في الطباعة. تحتسب قيم هذا النظام بالشكل التالي:

$$C = 255 - R$$

$$M = 255 - G$$

$$Y = 255 - B$$

- استحدث الفضاء اللوني CMYK بإضافة اللون الأسود إلى الفضاء اللوني

CMY لأهمية الحبر الأسود في الطباعة وصعوبة تشكيله. تحتسب قيم هذا النظام بتحديد أقل قيمة بين C، M وY وتدعى القيمة L ثم:

$$C = \frac{C - L}{255 - L}$$

$$M = \frac{M - L}{255 - L}$$

$$Y = \frac{Y - L}{255 - L}$$

$$K = \frac{L}{255}$$

- في الفضاء اللوني HSI: يمثل Hue درجة اللون، Saturation إشباع اللون وIntensity شدة الإضاءة.
- تقوم التنقية باستخدام الوسط الحسابي Mean Filtering بتعديل قيمة لون كل بيكسل باحتساب الوسط الحسابي لهذه البيكسل مضافاً إليها قيم البيكسلات المجاورة لها.
- تقوم التنقية باستخدام الوسيط الحسابي Median Filtering بتعديل قيمة لون كل بيكسل باحتساب الوسيط الحسابي لهذه البيكسل مضافاً إليها قيم البيكسلات المجاورة لها.
- تستخدم طريقة التنقية باستخدام الوسيط الحسابي في التخلص من شوائب الملح والبهار Salt and Pepper Noise، إلا أنها تؤثر على حدة الحواف في الصورة.
- تعطي التنقية باستخدام مصفوفة جاوس Gauss وزناً أكبر في حساب المعدل للبيكسل المتوسطة ومن ثم البيكسلات المجاورة لها بشكل أفقي وعمودي ومن ثم تلك المجاورة قطرياً.
- يهدف التعديل الخطي Linear Mapping إلى زيادة أو تقليل الإضاءة في الصورة بإضافة أو إنقاص قيمة ثابتة للقيم اللونية للبيكسلات في الصورة.

- عند إجراء التعديل الخطي للصور ذات التدرج الرمادي مثلاً فإن أية قيمة معدلة لا يمكن أن تتجاوز 255 أو تقل عن 0.
- التمثيل الرمادي Histogram هو رسم بياني يمثل المحور السيني فيه القيم اللونية المستخدمة في الصورة، ويمثل المحور الصادي تكرار هذه القيمة ضمن بيكسلات الصورة.
- يقوم التعديل الخطي بتحريك قيمة كل بيكسل في الصورة على التمثيل البياني بمقدار ثابت، وبالتالي تحريك التمثيل البياني للصورة يميناً أو يساراً.
- يمكن إجراء التعديل الخطي لواحدة أو أكثر من قنوات الفضاء اللوني Channels للصور الملونة، كزيادة أو إنقاص قيم اللون الأحمر في الفضاء اللوني RGB مثلاً.
- يقوم التعديل اللوغاريتمي Logarithmic Mapping بزيادة/تقليل مقدار متغير على القيمة اللونية للبيكسلات. ويعتبر التعديل اللوغاريتمي أفضل في الصور ذات الأجسام الداكنة على خلفية فاتحة وبالعكس.
- قلب الصورة Invert هو استبدال قيمة كل بيكسل بالقيمة المقابلة تماماً في التمثيل البياني، فتستبدل القيمة 0 في التدرج الرمادي بالقيمة 255، والقيمة 230 بالقيمة 20 وهكذا.
- يقوم قلب الصورة Invert بقلب التمثيل البياني للصورة أفقياً Flip Horizontally.
- الحافة في الصورة هي المنطقة التي يحدث فيها تغيير مفاجئ وحاد في القيمة اللونية للبيكسلات المتجاورة.
- تستخدم مصفوفات سوبل Sobel لتحديد الحواف الأفقية والعمودية في الصورة.
- قامت شركة Microsoft بتطوير ملف Bitmap (.bmp) للصور غير المضغوطة، ويحتوي ملف Bitmap على 14B ترويسة الملف، 40B ترويسة المعلومات، جدول الألوان (إن وجد) ومعلومات البيكسلات.

Chapman N. and Chapman J. (2000) Digital Multimedia. USA: John Wiley and Sons.

Efford, Nick. (2000) Digital Image Processing: A practical introduction using JAVA. England: Person Education Limited.

Jahne B. (2002) Digital Image Processing. Germany: Springer-Verlag Berlin Heidelberg.

Sonka M. et. al (1999). Image Processing, Analysis, and Machine Vision. USA: Brooks/Cole Publishing Company.

Vaughan Tay (2001) Multimedia – Making It Work. 5th Edition. USA: McGraw-Hill.

Villamil-Casanova J. and Molina L. (1997) Multimedia – An Introduction. US: Prentice Hall.

1. في أي من زوايا الصورة الأربعة ذات الحجم 200×200 تقع البيكسل (200، 0)؟
2. صورة ذات تمايز بمقدار 400 × 300 بيكسلاً وذات تمايز لوني بمقدار 8 bits:

   أ. احسب حجم الصورة بوحدة البايت B.

   ب. احسب عدد الألوان التي يمكن تمثيلها في هذه الصورة.

   ج. إذا تغير التمايز اللوني للصورة إلى 16 bits فما هو الحجم الجديد للصورة؟
3. إذا كان التمايز اللوني للصورة التالية هو 2 bits، وكانت الألوان الأربعة المستخدمة في الصورة هي: الأحمر، الأخضر، الأصفر والأسود، مرتبة حسب قيمها، أي أن الأحمر يمثل بالرمز 00، الأخضر بالرمز 01، الأصفر بالرمز 10 والأسود بالرمز 11.

   أ. مثل قيم الصورة التالية:

أسود	أخضر	أحمر
أصفر	أخضر	أحمر
أصفر	أحمر	أحمر

   ب. احسب حجم الصورة بوحدة Bit.
4. أكمل الجدول التالي بما يناسبه من قيم الألوان:

اللون	الأحمر	الأخضر	الأزرق	التمثيل السداسي
البرتقالي				
الأزرق الغامق				
الرمادي الفاتح				
				#AA806F
				#30402B
	112	112	65	

5. اشرح السبب وراء ظهور الفضاء اللوني CMY.
6. اشرح السبب وراء ظهور الفضاء اللوني CMYK.
7. اشرح السبب وراء ظهور الفضاء اللوني HSI.
8. إذا كان الفضاء اللوني RGB مستخدما في تمثيل صورة معينة، وكان التمثيل الرقمي التالي يمثل معلومات الصورة، أجب عن الأسئلة التالية:

0010100101111110000101101011010110000000111111111

أ. كم عدد البيكسلات الممثلة؟

ب. حول قيمة البيكسل الثانية إلى النظام السادس عشر.

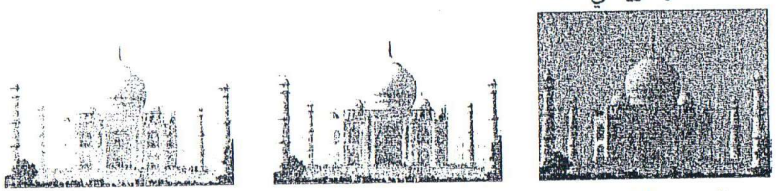
ج. هل تستطيع تقدير اللون الحقيقي للبيكسل الثانية؟ (استخدم نظام Photoshop للتأكد من استنتاجك)

9. حول القيمة #AB9C34 من الفضاء اللوني RGB إلى الفضاء اللوني CMYK.
10. حول القيمة #B2DF92 من الفضاء اللوني RGB إلى الفضاء اللوني YUV.
11. حول القيمة CMY = (123 231 90) إلى التمثيل السادس عشر للفضاء اللوني RGB.
12. ما هي النقاط التي يجب أن تستفسر عنها عند شرائك ماسحة ضوئية جديدة؟

13. ما هي قطعة CCD ولماذا نحتاجها في الماسحات الضوئية والكاميرات الرقمية؟
14. ما هو الفرق بين شاشات العرض CRT وشاشات عرض LCD؟
15. ارسم التمثيل البياني Histogram للصورة ذات البيكسلات التالية: (لاحظ أن التمايز اللوني للصورة هو 2 bits/pixel. ثم أجر تعديلاً خطياً بمقدار 1 وارسم التمثيل البياني الجديد.

00	11	10	01	01
00	00	10	10	10
10	00	01	10	10
11	00	01	01	10
11	01	01	01	01
00	01	01	01	01

16. الصورة التالية تم تعديلها باستخدام التعديل الخطي والتعديل اللوغاريتمي. أي الصورتين المعدلتين تنتمي للتعديل الخطي وأيهما تنتمي للتعديل اللوغاريتمي؟ لماذا؟



الصورة الأصلية | الصورة المعدلة1 | الصورة المعدلة2

الصورة: تاج محل – أجرا – الهند (2004)

17. في الصورة التالية ذات التدرج الرمادي

123	123	123	123	123
124	124	124	124	124
244	28	28	244	244
245	16	12	244	244
244	244	244	244	244
244	244	244	244	244

أ.قم بتنقية الصورة باستخدام الوسط الحسابي

ب.قم بتنقية الصورة باستخدام الوسيط الحسابي

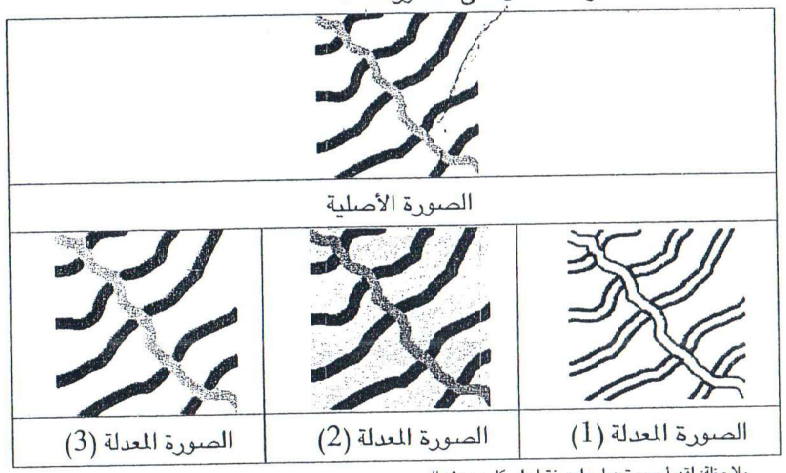
ج.قم بتنقية الصورة باستخدام مصفوفة Gauss

د.أي الطرق التالية كانت أكثر نجاعة في التخلص من الشوائب المظللة.

18. باستخدام مصفوفتي سوبل، قم بتحديد الحواف العمودية والأفقية في الصورة التالية علما بأن الفضاء اللوني المستخدم فيها هو التدرج الرمادي

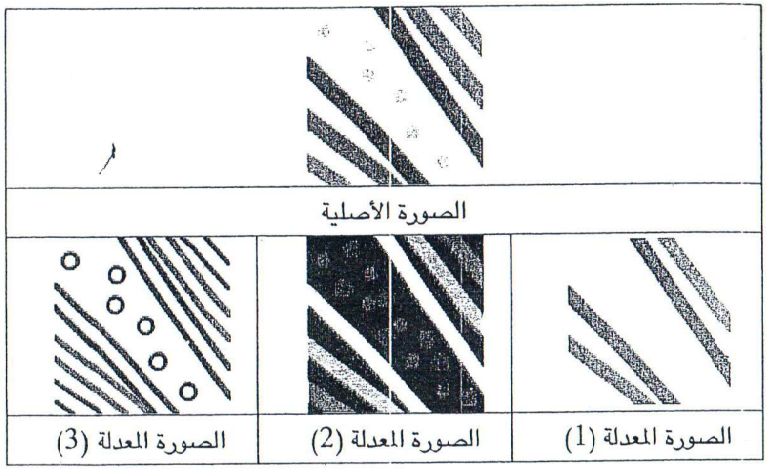
123	123	123	123	123
124	124	124	124	124
244	244	244	244	244
245	244	244	244	244
244	244	244	244	244
244	244	244	244	244

19. ما هو التغير الحاصل على الصور المعدلة؟

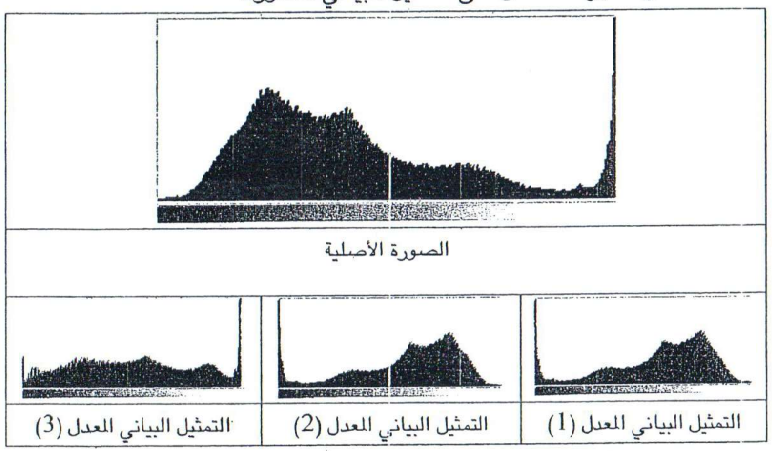


ملاحظة: لقد أجري تعديل واحد فقط على كل من هذه الصور.

20. ما هو التغير الحاصل على الصور المعدلة؟



21. ما هو التغير الحاصل على التمثيل البياني للصورة؟



22. كيف وأين تخزن المعلومات التالية عن الصورة في ملف Bitmap:

أ. تمايز الصورة

ب. التمايز اللوني

ج. حجم الملف

د. طريقة الضغط

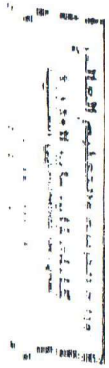
23. اكتب برمجية تقوم على قلب Invert الصورة باستخدام لغة Java.

24. حول القيمة التالية للبيكسل إلى النظام السادس عشر، ثم حاول اكتشاف اللون الممثل لهذه القيم.

01111111011110001011110 1

25. ما هو التمثيل الرقمي للبيكسل من النظام السادس عشر التالي: #A8450F؟

26. حاول تمثيل الألوان التالية باستخدام الفضاء اللوني RGB: الأخضر الداكن، الرمادي الفاتح، الأصفر الداكن، البرتقالي الداكن، الأزرق السماوي.



الوحدة الرابعة

الرسوم المتحركة

Animation