



جامعة دمشق

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الإلكترونيات والاتصالات

أتمتة خدمات مشفى

تصميم وتنفيذ نظام خدمات مشفى باستخدام انترنت
الأشياء

مشروع تخرج أعدد لنيل درجة الإجازة في هندسة الإلكترونيات والاتصالات

إعداد الطالب:

محمد مطيع الصابوني

بإشراف

د. طلال الحمود

العام الدراسي

2021-2020

شكر وتقدير

بسم الله الرحمن الرحيم

وأفضل الصلاة وأتم التسليم على سيدنا محمد خاتم المرسلين... الحمد لله الذي علّمنا ما لم نكن نعلم... اللهم علّمنا ما ينفعنا وانفعنا بما علّمتنا وزدنا علماً.

إلى كلّ من زرع التفاؤل في دربي وأنار طريقي بالعلم والمعرفة وإلى الذين لم يوفّروا جهداً حقيقياً صادقاً معي إلا بذلوه فكان ذلك الحافز القوي الذي أعطاني همة عالية أبذلها لكي أكون ثمرة حقيقية. راجياً الله أن يجعلها في ميزان حسناتهم فلهم منّي كلّ الاحترام والتقدير.

وأخصّ منهم بالشكر:

الدكتور المهندس طلال الحمود

الذي وجّهني ومدّ لي يد العون والمساعدة وكان معي خطوة بخطوة في سبيل إنجاز هذا العمل على أكمل وجه ... جزاه الله عنّي كلّ خيرٍ وجعلها في ميزان حسناته.

المحتويات

5	جدول أهم الرموز الاصطلاحية المستخدمة
6	مقدمة عامة
9	القسم الأول- انترنت الأشياء:
9	مقدمة تمهيدية:
10	بداية انترنت الأشياء:
11	بنية أجهزة انترنت الأشياء:
12	مراحل عمل تقنية انترنت الأشياء:
13	مميزات إنترنت الأشياء:
14	شريحة الربط مع الانترنت ESP:
17	الفصل الثاني – شريحة الـESP:
17	مقدمة عامة:
18	شرح أطراف شريحة ESP32:
21	إصدارات الـESP:
22	مميزات شريحة الـESP:
27	القسم الثالث- التعلم الآلي:
29	بعض أنواع خوارزميات التعلم:
32	خوارزمية آلة المتجه الداعم <i>Support vector machine</i> :
36	أنواع متجهات دعم الآلة:
40	إيجابيات وسلبيات خوارزمية آلة المتجه الداعم <i>SVM</i> :
41	الرؤية الحاسوبية:
41	تطبيقات الرؤية الحاسوبية:
42	المهام الرئيسية للرؤية الحاسوبية:
43	معالجة الصورة الرقمية <i>Digital Image processing</i> :
44	أساسيات معالجة الصورة <i>Image Processing</i> :
47	خوارزمية فيولا جونز:
53	AdaBoost:
54	Cascade Classifier:

56 hog : خصائص hog
57 الجزء العملي
58 مقدمة:
59 المخطط الصندوقي للمشروع:
60 معلومات الكادر الطبي:
61 الربط مع الانترنت:
61 الرؤية الحاسوبية:
63 البيئة البرمجية Pycharm باستخدام لغة Python:
64 شرح أجزاء المشروع:
64 شريحة التحكم ESP32:
66 القارئ المستخدم RC522 RFID:
67 التوصيلات الالكترونية للشريحة ESP32:
73 منصة IOT المنفذة:
75 آلية العمل والمخطط التدفقي لكود شريحة الربط مع الانترنت:
75 تطبيق أندرويد المنفذ :
79 المكتبات المستخدمة:
79 مكتبة scikit-learn (SK learn):
80 مميزات مكتبة scikit-learn :
81 مكتبة Opencv:
84 آفاق مستقبلية للمشروع.....
84 خاتمة.....
85 ملحق -الكودات البرمجية الأساسية.....
85 كود حساس الحرارة TMC 103:
88 كود الـ RFID:
91 كود حساس القلب والأكسجة Max30100:
93 كود تحليل الفيديو والمراقبة:
94 الكود الثاني:
96 كود الصورة:
98 المراجع:

جدول أهم الرموز الاصطلاحية المستخدمة

العبارة الكاملة بالإنكليزية، والمقصود بها	رمز المصطلح Acronym	
Internet of Things	IOT	1
Internet of Everything	IOE	2
Wireless Fidelity	WI-FI	3
Integrated Circuit	IC	4
Institute of Electrical and Electronics Engineers	IEEE	5
Universal Serial Bus	USB	6
Universal Asynchronous Receiver/Transmitter	UART	7
Inter-Integrated Circuit	I2C	8
Communication port (serial)	COM	9
Serial Peripheral Interface	SPI	10
Infra-Red	IR	11
Radio Frequency	RF	12
Transistor transistor Logic	TTL	13
Pulse Width Modulation	PWM	14
General Purpose Input Output	GPIO	15
Integrated Development Environment	IDE	16

مقدمة عامة

لقد استخدم الإنترنت وحتى وقت قريب "بداية التسعينات الميلادية" من قبل الأكاديميين والأشخاص المهتمين بتقنية المعلومات والحاسب بشكل عام.

ولكن في الوقت الحاضر ازداد الاهتمام بالإنترنت على نطاق واسع خاصة في المجال التجاري والاستثمارات التجارية والمجال الصحي.

وقد حولت تقنية المعلومات طريقة ومخطط عمل الخدمات الصحية وساعدت على تقديم خدمات بطريقة أفضل وأسرع.

وتعتبر العلاقات مع العملاء والمرضى القوة المحركة لهذا التغير.

تتمثل أهمية أتمتة خدمات المشفى في أنها تأتي استجابة للتطور السريع في عالم الاتصالات والتوجه نحو الأعمال الإلكترونية في كافة الأنشطة الإنسانية التجارية منها والصناعية والصحية، فالتوجه نحو الأعمال الإلكترونية بات ضرورة حتمية على المجتمعات في كافة القطاعات، ومن بين هذه القطاعات القطاع الصحي، كما أن الصحة الإلكترونية انتشرت انتشاراً كبيراً في الدول المتقدمة، مما يتطلب دراسة واقع استخدام الصحة الإلكترونية في المستشفيات السورية.

يعاني النظام التقليدي في المشافي من العديد من العيوب والسلبيات أبرزها:

- الاختلاط والتماس المباشر مع المريض.
- العدوى وانتقال الأمراض نتيجة الازدحام الذي يحدث في المشافي.

كما تتمثل أهمية أتمتة خدمات المشفى في تقديمه لأهم الأعمال الإلكترونية التي يمكن أن تستخدم في المستشفيات، وأسباب استخدامها:

- 1) المساعدة في تحديد دور الأعمال الإلكترونية في تطوير ومساعدة المستشفيات للقيام بالأعمال بسلاسة وبسر داخلياً وخارجياً.
- 2) المساعدة في تجاوز العقبات للارتقاء بالخدمات في المجال الصحي وتحفيز المدراء على تطبيق الأعمال الإلكترونية عند معرفة الإيجابيات والسلبيات وكيفية التطبيق.
- 3) التعرف على مدى ق درة نظام الصحة الإلكترونية على تحسين العلاقة بين المستشفيات وعمالها المرضى وشركات التأمين وشركات الأدوية والصيدليات وكذلك المعاهد الطبية والمؤسسات التنظيمية الحكومية.

و تتمثل إحدى أبرز أتمتة خدمات المشفى في الحفاظ على سلامة الأشخاص والمباني وأمنها. لهذا السبب، نوّقر بنية تحتية وأنظمة ذكية موثوق بها وممكنة لإنترنت الأشياء تجعل الطاقة والتقنيات الفعالة متاحة بشكل مستمر طوال فترة الرعاية.

يعتمد النظام المقترح على تحقيق الأفكار التالية:

- استخدام بطاقة من النوع RFID كبطاقة ذكية لدخول الممرضين إلى غرف المرضى.
- الربط مع منصة إنترنت الأشياء, للتحقق من صحة المريض.
- الحرص التام على ارتداء الكمامة عند الدخول إلى المشفى وذلك من خلال كاميرات ذكية.

يدرس الفصل الأول تقنيات إنترنت الأشياء والأفكار النظرية والعملية المرتبطة بها.

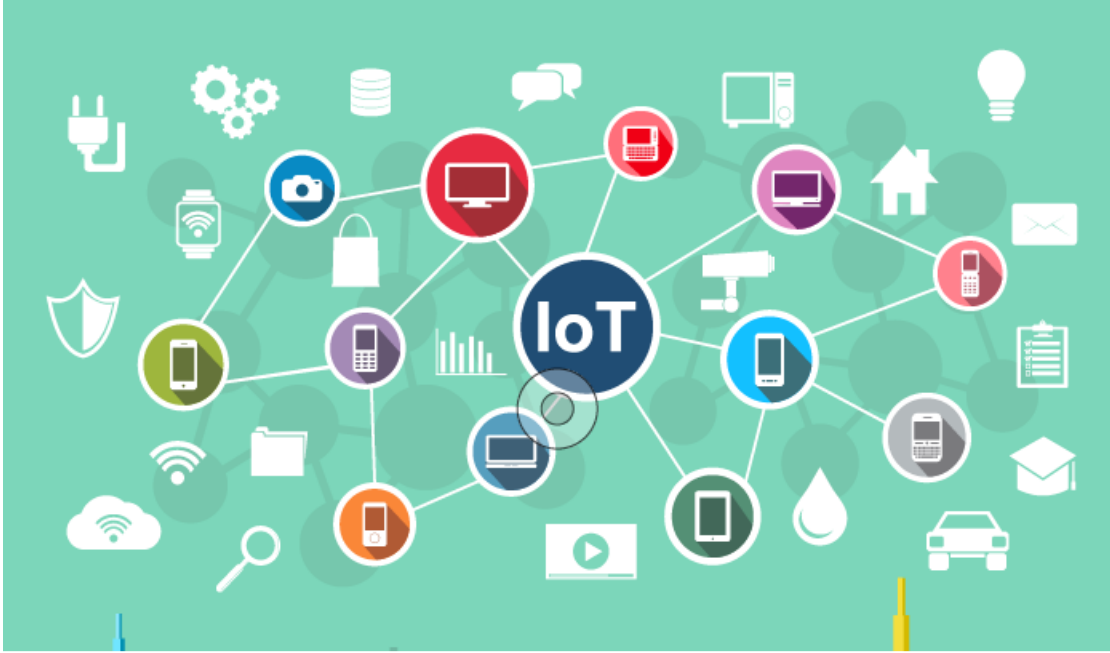
ويتناول الفصل الثاني الجزء العتادي الإلكتروني للمشروع ويشرح عن شريحة الـ ESP32.

ويدرس الفصل الثالث عن التعلم الآلي و كيفية توظيفه في مشروعنا المستخدم.

أما في الفصل الرابع فيتحدث عن الجزء العملي للمشروع و كيفية توظيف كل قسم مع بعضه البعض ليشكل نظاماً متكافلاً جديراً بالثقة و الاستخدام.

الفصل الأول إنترنت الأشياء

Internet of things IOT



القسم الأول - انترنت الأشياء :

إنّ انترنت الأشياء Internet Of Things هو مفهوم متطور لشبكة الإنترنت بحيث تمتلك كل الأشياء قابليّة الاتصال بالإنترنت أو ببعضها البعض لإرسال واستقبال البيانات لأداء وظائف محددة من خلال الشبكة.

مقدمة تمهيدية:

حاول البشر منذ القدم اختراع أدوات تساعدهم في تسريع عمليّاتهم الحسابيّة والتحليليّة، كانت تلك الأدوات بدائيّة في بعض الأحيان ولكن كانت بالنسبة لتلك الأزمان ذات تطور لافت لأنّها تمكّنت من تسريع بعض الأمور التي بدت وكأنّها معقّدة ويحتاج الإنسان لوقت طويل إمّا لحسابها أو للتعامل معها، ولكن تلك المحاولات بقيت بسيطة حتى القرن العشرين عندما بدأت محاولات اختراع الحواسيب إلى أن أصبحت المعالجات في وقتنا الحالي تنفّذ الملايين من العمليات الحسابيّة في وقت قصير فظهرت الحواسيب المحمولة والأجهزة اللوحية.

إنّ هذه الحواسيب قادرة تشغيل أي نوع من الأكواد والقيام بعدد هائل من المهمّات ولكن قد لا نحتاج إلى هذه القدرات الكبيرة في أغلب الأحيان فهي أجهزة ذات أغراض عامّة يمكنها القيام بأي مهمّة بجودة مقبولة، وهذا هو سبب ارتفاع سعرها برغم من عدم الحاجة إلى معظم إمكانيّاتها.

من هنا أتت الحاجة إلى أجهزة ذات أغراض خاصّة تقوم بمهام معينة فقط بالتالي لا يكون هناك هدر في القدرات بالإضافة إلى انخفاض التكاليف والأهم من ذلك الجودة والكفاءة العالية في تنفيذ المهام لأنّ تصميم هذه الأجهزة من الناحية البرمجية أو العتادية يكون متمحوراً حول القيام بمهمّة واحدة بالتالي كفاءة وجودة تنفيذ.

بإضافة الاتصال بالإنترنت إلى هذه الأجهزة الذي أصبح سهلاً جداً في يومنا هذا تمّ التوصل إلى تقنية جديدة سمّيت بإنترنت الأشياء التي تجمع بين ميزات أجهزة الأغراض الخاصة والإمكانيات الهائلة لشبكة الإنترنت.

بداية انترنت الأشياء:

○ في عام 1982 قام مجموعة من العلماء الذين كانوا يعملون في إحدى الطوابق المرتفعة في إحدى مباني جامعة KING بربط ماكينة المشروبات الغازية الموجودة في الطابق السفلي بشبكة المبنى الداخلية ليتمكنوا من معرفة إذا كانت هنالك علبة مشروبات غازية في الماكينة أم أنّها فارغة حتى لا يضطروا للنزول من دون فائدة.

○ في عام 1999 ولأوّل مرة تمّ ذكر مصطلح انترنت الأشياء عن طريق Kevin Ashton الذي كان يلقي محاضرة في إحدى المؤتمرات وقام بطرح هذا المصطلح ليعبر عن فكرة في باله وربط الأشياء والآلات بالإنترنت ثمّ بدأت هذه العجلة بالدوران وبدأت الشركات بالتسابق والتفكير ببناء منتجات تحت مفهوم انترنت الأشياء.

○ في عام 2005 قام الاتحاد الدولي للاتصالات ITU بطرح أول ورقة بحثية عن انترنت الأشياء والتي أوضحت رؤيته والتكنولوجيا التي تُستخدم ببنائه وأوضحت أيضاً أهم التحديات التي تواجهها واستخداماتها المستقبلية وفوائدها.

○ هناك أيضاً من يستخدم مصطلحات بديلة مثل اتصال الآلات M2M، انترنت الأشياء IOT، انترنت كل شيء IOE.

بنية أجهزة انترنت الأشياء:

لمعرفة معنى وتعريف انترنت الأشياء فنجد أنّ عدّة تعاريف ظهرت من عدّة جهات لها فهناك من ركّز على الاتصال بالإنترنت ومنهم من ركّز على البنية التحتية وآخرين ركّزوا على الاتصالات بين الأجهزة المختلفة، من تلك الجهات IEEE, ITU, Cisco, 3GPP.

ولو أردنا تلخيص تلك التعاريف وفهم المعنى الحقيقي له فنستطيع القول: إنّها الأشياء الحاوية على الذكاء الحاسوبي والقابلية للاتصال والمتصلة بالإنترنت، مُشكّلةً شبكة من الأشياء القادرة على التفاعل مع محيطها والتكلم مع بعضها ومع الأجهزة والأنظمة المتصلة بالإنترنت. هذه التقنية ليست فقط لوصول الأشياء والآلات بل وصلت الخدمات والإجراءات والعمليات وأتمنتها من خلال منصّة اتّصال مثبتة على الانترنت ليكون أساساً للمجتمع والاقتصاد الرقمي.

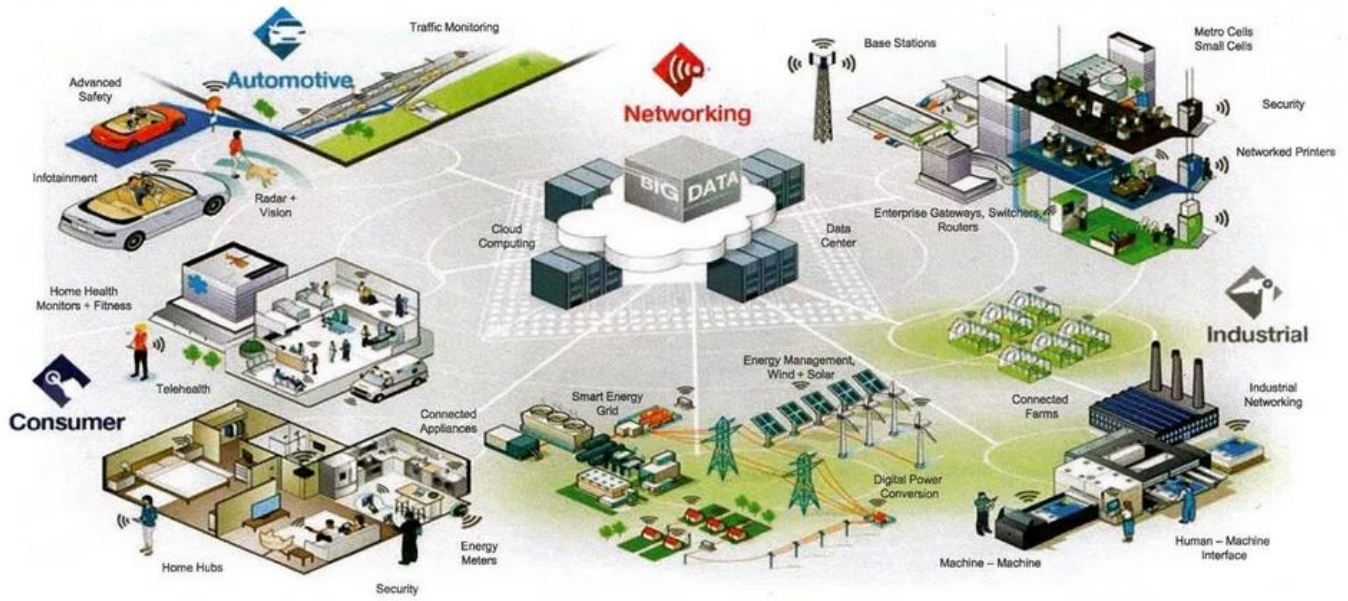
هناك عدّة خطوات تحوّل تلك الأشياء من تقليدية إلى أشياء انترنت وأهمها:

- إضافة تعريف للشيء وذلك عن طريق عنوان محدد أو فريد بكل شيء عن عناوين نظيراتها.
- إعطاءها قدرة على الإحساس، التي تمكننا من جمع المعلومات والمتغيرات في البيئة التي تتواجد فيها وإرسالها إلى مكان آخر.
- تزويدها بالقدرة على معالجة تلك المعلومات التي تمّ تجميعها وأخذ بعض القرارات عن طريق تضمين الذكاء الحاسوبي فيها.
- تزويدها بقدر معين من الأمان لحمايتها من الهجمات أو الفيروسات وذلك يتم بطرق مختلفة من طرق حماية المعلومات والشبكات.

- يجب أن تكون الأشياء تعمل بأقل طاقة ممكنة وذلك لتمكّنها من العمل مدّة طويلة وخصوصاً أن تلك الأشياء متوقّع منها العمل باستمرار وفي حالات معينة في أماكن بعيدة عن مصادر الطّاقة.
- أن تكون ذات تكلفة منخفضة وخصوصاً لاحتياجنا لوجودها بكميات كبيرة لتمكّنا من وصل عدد أكبر من الأشياء وجعل هذه الشّبكة من انترنت الأشياء شاملة قادرة على جمع العمليات والمهام في حياتنا كلها.
- ذات جودة واعتمادية عالية حتى نستطيع الوثوق بها والاعتماد على المعلومات والقرارات التي تزودنا بها.

مراحل عمل تقنية انترنت الأشياء:

- مرحلة تحصيل البيانات: يتم تحصيل البيانات من البيئة الفيزيائية المحيطة من قبل الحساسات التي زودت بها أجهزة النظام.
- مرحلة إرسال البيانات إلى سحابة التخزين: يتم إرسال البيانات التي تم تحصيلها من الوسط الفيزيائي إلى العالم الرقمي عبر Getaway (موجه Router) الذي يؤمن اتّصال أجهزة النّظام بشبكة الانترنت.
- مرحلة معالجة البيانات وتخزينها: تتم معالجة البيانات وتخزينها بواسطة المخدمات السحابية Cloud Servers التي تشكّل بنكاً للمعلومات المحصلة يمكن الوصول إليها من كل أنحاء العالم.
- مرحلة الوصول إلى البيانات وعرضها: توفّر المخدمات السحابية أيضاً منصّات وواجهة مستخدم يتم فيها عرض المعلومات بأشكال وطرق مختلفة سهلة القراءة وتساعد في اتخاذ القرارات المناسبة (متوسّطات حسابية خلال فترات زمنية معينة، مخططات بيانية، القيم العظمى والدنيا إلخ)
- يتم الوصول إليها من خلال شاشة الحاسوب أو الهاتف الذكي والأجهزة اللوحية المشابهة.



الشكل (1-1) تطبيقات وانتشار انترنت الأشياء IOT

مميزات إنترنت الأشياء :

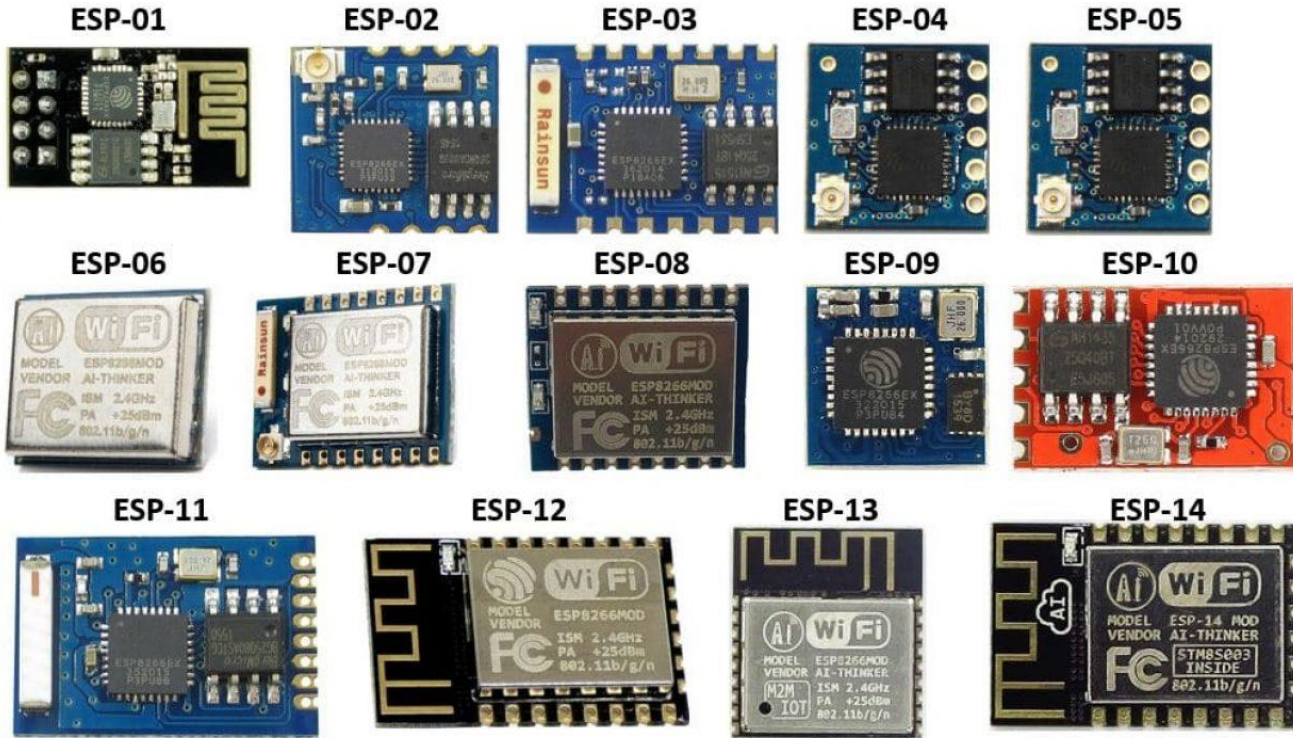
- إنترنت الأشياء سوف يجعل أداء المهام أكثر سهولة بشكل كبير .
- باستخدام تكنولوجيا إنترنت الأشياء يمكن القيام بكثير من المهام عبر الجوال ويمكن برمجتها لكي تعمل بصورة آلية تماماً بدون أي تدخل يدوي .
- إنترنت الأشياء سوف يوفر الكثير من الوقت للجميع .
- إنترنت الأشياء يعمل في اتجاه توفير الطاقة .
- الأجهزة والأشياء الذكية في كل مكان تعمل فقط عند الضرورة .

○ انترنت الأشياء سوف يجعلنا نتمتع بالمزيد من الوقاية والرعاية الصحية.

مثلاً في حالة سيارة ذكية متصلة بالإنترنت، تقوم بالاتصال بالإسعاف بمجرد حدوث أي تصادم يعرض راكبها لمخاطر صحية.

شريحة الرّبط مع الانترنت ESP:

- وهي وحدة Wi-Fi شائعة الاستخدام في مشاريع انترنت الأشياء، حيث تسمح هذه الوحدة للمتحكمات الصغيرة بالوصول إلى شبكة Wi-Fi وهي عبارة عن SOC (نظام على شريحة System On a Chip) أي أنها لا تحتاج بالضرورة إلى متحكم لمعالجة المدخلات والمخرجات، لأنها تعمل كحاسب صغير. وتبعاً لإصدار ESP، من الممكن أن تحوي هذه الوحدة على ما يصل إلى 9 GPIOs (مخرجات إدخال الأغراض العامة)، ويمكننا ببساطة برمجة ESP ليس فقط للوصول إلى شبكة Wi-Fi، ولكن أيضاً للعمل كمتحكم. هذا ما يجعل ESP ذو خيارات متنوعة، وتوفّر المال والمساحة في مشاريع انترنت الأشياء. وتتوفّر بعدة أنواع.



الشكل (1-2) أنواع شريحة الربط مع الانترنت

- تمكّننا هذه الشريحة من الاتصال بموقع ما على شبكة الانترنت والتواصل معه.
- يجب عند استخدام الشريحة ESP للمرة الأولى التأكد من الإعدادات الصحيحة لها من خلال التخابط معها في وضعية الإعدادات AT Command والتي يتم الدخول إليها من خلال وصل المدخل GPIO0 إلى الأرضي عند بدء التشغيل ثم وصله إلى التغذية.
- إنّ جهد التغذية الاعظمي لها 3.6V حيث أنّها تملك ضمنها منظّم يضبط جهد التغذية يجعله 3.3V , بالتالي فإنّ مداخلها و مخرجها I/O Pins تعمل على 3.3V إذ سنحتاج إلى متحكم بمستوى الجهد المنطقي عند إدخال أي إشارة 5V إلى الشريحة . كما يمكن بدلا من ذلك استخدام منظّم خارجي.

الفصل الثاني

شريحة

ESP32



الفصل الثاني - شريحة الـ ESP:

ESP32 عبارة عن سلسلة من المتحكمات الصغيرة منخفضة التكلفة ومنخفضة الطاقة، والتي تعتمد في تصنيعها على تقنية (نظام على رقاقة System on Chip) مع تقنية Wi-Fi مدمجة وبلوتوث مزدوج الوضع.

مقدمة عامة:

في الربع الأخير من عام 2016 تم الإعلان عن ESP32 في عالم النظم المضمنة المزدهم بالكثير من الحلول والدارات الجديدة. تستخدم سلسلة ESP32 معالجاتاً دقيقاً اسمه Tensilica Xtensa LX6 في كل من النوعين ثنائية النواة ووحيدة النواة، وتتضمن مفاتيح هوائي مدمجة ، ومضخم الترددات اللاسلكية ، ومضخم الطاقة ، ومضخم الاستقبال منخفض الضوضاء ، والمرشحات ، ووحدات إدارة الطاقة. تم إنشاء وتطوير ESP32 بواسطة Espressif Systems . وهي شركة صينية مقرها في شنغهاي . ويتم تصنيع الشريحة بواسطة شركة TSMC باستخدام عملية الـ 40 نانومتر. إنها خليفة المتحكم الصغير SESP8266.

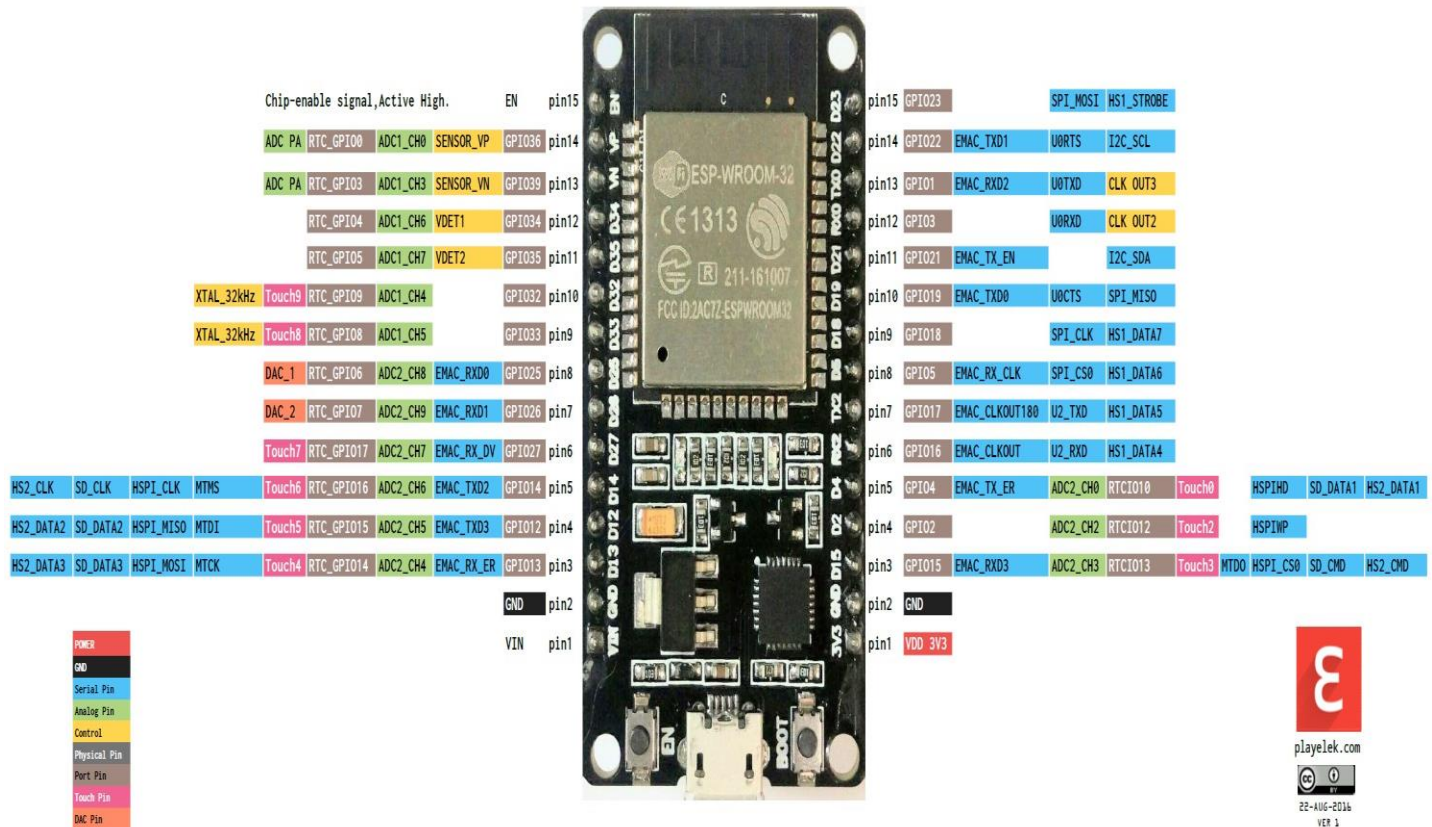


ESP32 هي SoC تحوي طرفيات غنيّة جداً، حيث تحصى الـ Datasheet أكثر من 19 طرفية داخل هذه الـ IC (الدائرة المتكاملة) إلى جانب طرفيتي البلوتوث والواي فاي. ليس هذا فقط، بل أيضاً تحوي ESP32 (في نسخ محدّدة منها) على نواتي معالجة dual core 32-bit microprocessor.

إن شريحة ESP32 هي الأخ الأصغر للـ ESP8266، وهي تتميز بسعرها الرخيص وقدرتها على الاتصال بالإنترنت والبلوتوث، كما أنها قطعة سهلة البرمجة بل ويمكننا برمجتها بواسطة بيئة تطوير الأروينو (Arduino IDE) كما سنرى لاحقاً.

متطلبات الطاقة: نظراً لأن نطاق جهد التشغيل لـ ESP32 يتراوح من 2.2 فولت إلى 3.6 فولت، فإن اللوحة تأتي مع منظم جهد LDO للحفاظ على الجهد ثابتاً عند 3.3 فولت. يمكن أن توفر ما يصل إلى 600 ميلي أمبير بشكل موثوق، والتي ينبغي أن تكون أكثر من كافية عندما تسحب ESP32 ما يصل إلى 250 ميلي أمبير أثناء نقل الترددات اللاسلكية. يتم تقسيم خرج المنظم أيضاً إلى أحد جانبي اللوحة. يمكن استخدام هذا الدبوس لتوفير الطاقة للمكونات الخارجية.

شرح أطراف شريحة ESP32:



Power Pins وهي دبابيس الطاقة. هناك اثنين من دبابيس الطاقة بمعنى دبوس Vin و 3.3 فولت يمكن استخدام دبوس Vin لتزويد ESP32 وملحقاته مباشرة بالطاقة إذا كان لديك مصدر جهد 5 فولت منظم . دبوس 3.3 فولت هو إخراج منظم الجهد على متن اللوحة. يمكن استخدام هذا الدبوس لتوفير الطاقة للمكونات الخارجية.

GND هو دبوس الأرضي للـ ESP32.

Arduino Pins وهي دبابيس للأردوينو وهي ليست سوى دبابيس لبروتوكولي I²C و SPI في ESP32 لتوصيل جميع أنواع المستشعرات والأجهزة الطرفية في المشروع .

GPIO Pins تحتوي اللوحة التطويرية ESP32 على 25 دبوس تستخدم للإدخال والإخراج التي يمكن تعيينها لمختلف الوظائف برمجياً. يمكن تهيئة كل GPIO مزود بتقنية رقمية لسحب أو سحب داخلي ، أو ضبطه على مقاومة عالية. عند تهيئته كمدخلات ، يمكن أيضاً ضبطه على مشغل الحافة أو المشغل على مستوى، لإنشاء مقاطعات وحدة المعالجة المركزية.

ADC Channels يتم دمج 18 محول تماثلي رقمي في اللوحة ADC 12-bit (تكون دقة المحولات فيها 12 بت) وتدعم القياسات على 15 قناة (دبابيس تمكين تماثلية) يمكن استخدام بعضها في بناء مضخم ربح قابل للبرمجة يستخدم لقياس الإشارات التناظرية الصغيرة. تم تصميم ESP32 أيضاً لقياس الفولتية أثناء التشغيل في وضع السكون.

نستطيع حساب قيمة أصغر تغير في الجهد يستطيع الـ ESP32 قراءته باستعمال العملية الحسابية التالية:

$$(2^{12} - 1) / 3.3 = 4095 / 3.3 = 81 \text{ mV}$$

DAC Channels تحتوي اللوحة على قناتين DAC من 8 بتات لتحويل الإشارات الرقمية إلى فولتات تمثيلية حقيقية. يمكن لهذا الـ DAC المزدوج أن يستخدم في عدة تطبيقات مثل تعديل عرض النبضة PWM ، ويمكن أن يقود الدوائر الأخرى. أرقام الأطراف هي:

DAC_1 – GPIO26 و DAC_1 – GPIO25

Touch Pads وهي وسادات تعمل باللمس توفر اللوحة 10 أطراف GPIOs حساسة للسعة والتي تكشف عن الاختلافات في السعة التي أدخلت من جهة اتصال GPIO المباشرة.

External SD Card Pins تدعم شريحة الـ ESP إضافة بطاقة ذاكرة خارجية

UART Pins تحتوي اللوحة التطويرية ESP32 على بروتوكولات UART ، أي UART0 و UART2 ، والتي توفر اتصالاً غير متزامن (RS232 - RS485) ودعم IrDA ، وتتواصل بسرعة تصل إلى 5 ميجا بت في الثانية. يوفر UART إدارة الأجهزة لإشارات CTS و RTS والتحكم في تدفق البرامج (XON و XOFF) أيضاً.

I²C Pins تُستعمل هذه الأطراف من أجل عملية الاتصال أي سكويرد سي وهي تقنية تسمح بتشغيل وتبادل المعلومات مع الأجهزة باستعمال اشارتين كهربائيتين فقط.

I2S Pins إذا كنت ترغب في إضافة صوت إلى المشروع.

SPI Pins هي دبائيس SPI ESP32 تضم ثلاثة SPIs وهي (SPI – HSPI – VSPI) في أوضاع العميل والمخدم. تدعم SPIs أيضاً ميزات SPI للأغراض العامة التالية:

أربعة أوضاع توقيت لنقل تنسيق SPI .

ما يصل إلى 80 ميغاهيرتز والساعات مقسمة من 80 ميغاهيرتز .

ما يصل إلى 64 بايت FIFO .

يمكن أيضاً استخدام جميع SPIs للاتصال بـ Flash / SRAM وشاشة LCD الخارجية.

PWM Pins تحتوي اللوحة على 25 قناة تقريباً (جميع دبائيس GPIO) من دبائيس PWM التي يتم التحكم فيها بواسطة جهاز التحكم في تعديل عرض النبضة (PWM). يمكن استخدام خرج الـ PWM لقيادة المحركات الرقمية والمصابيح. تتكون وحدة التحكم من مؤقتات PWM ومشغل PWM . أقصى تردد عرض نبضة ممكن أن تولده هذه الشريحة 80 ميغا هرتز .

RTC Pins تُستعمل هذه الأطراف لإيقاظ الـ ESP من وضع الاستعداد (sleep mode).

EN Pin الموجود في الزاوية اليسرى العليا هو زر إعادة الضبط ، وتستخدم بالطبع لإعادة تعيين رقاقة الـ

ESP32 .

BOOT Pin في الركن الأيسر السفلي هو زر التنزيل المستخدم أثناء تنزيل البرامج الجديدة.

إصدارات الـ ESP:

Ordering code	Core	Embedded flash	Connection	Package
ESP32-D0WDQ6	Dual core	No embedded flash	Wi-Fi b/g/n + BT/BLE Dual Mode	QFN 6*6
ESP32-D0WD	Dual core	No embedded flash	Wi-Fi b/g/n + BT/BLE Dual Mode	QFN 5*5
ESP32-D2WD	Dual core	16-Mbit embedded flash (40 MHz)	Wi-Fi b/g/n + BT/BLE Dual Mode	QFN 5*5
ESP32-S0WD	Single core	No embedded flash	Wi-Fi b/g/n + BT/BLE Dual Mode	QFN 5*5

من الجدول السابق نجد أن إصدارات الـ ESP32 تختلف بالتالي:

1. عدد الأنوية cores.

2. وجود ذاكرة فلاش 16-Mbit داخل الدارة المتكاملة.

3. حجم غلاف الدارة المتكاملة (IC).

اسم الوحدة	اسم النسخة	الذاكرة الخارجية (4MB)	الهوائي(ات)
ESP32-WROOM-32	ESP32-D0WD	Flash	PCB
ESP32-WROOM-32D	ESP32-D0WD	Flash	PCB
ESP32-WROOM-32U	ESP32-D0WD	Flash	IBEX
ESP32-WROVER	ESP32-D0WD	Flash & SRAM	PCB
ESP32-WROVER-I	ESP32-D0WD	Flash & SRAM	PCB & IBEX
ESP32-SOLO-1	ESP32-S0WD	Flash	PCB

نجد من الجدول أعلاه (حتى هذا الوقت) أن جميع الوحدات تستخدم نفس رقم الـ ESP32 (النسخة) والذي هو ESP32-D0WD باستثناء ESP32-SOLO-1 الذي يستخدم ESP32-S0WD (وحيد النواة) ويبقى الشيء الذي تتمايز فيه:

1. نوع الهوائي.

2. وجود ذاكرة رام حجم 4 ميغا بايت من عدمه.

إن آخر شكل للـ ESP32 يمكن الحديث عنه هو ESP32-PICO-D4 وهو نظام داخل غلاف أو كما يعرف (System-in-Package (SiP حيث أنه يحوي الـ ESP32 إضافةً إلى كل العناصر المطلوبة من مكثفات الفلترة والذاكرة فلاش والمهتز وتقريباً كل شيء ما عدا الهوائي.

مميزات شريحة الـ ESP:

توفر ESP32 خيارين للاتصال اللاسلكي وهما: الواي فاي 802.11 b/g/n والبلوتوث نسخة 4.2 كما أن مواصفات المعالج والذاكر المتاحة أمر مهم حيث يتوفر Xtensa single-/dual-core 32-bit LX6 microprocessor(s) يعمل بسرعة 160 أو 240 ميغاهرتز وينفذ ما يصل إلى 600 DMIPS ، بالإضافة إلى ذواكر داخلية ROM بحجم 448 كيلوبايت و SRAM بحجم 520 كيلوبايت. ويدعم ESP32 ذواكر خارجية بحجم 16×4 ميغابايت عبر واجهة QSPI. يحوي ESP32 ذاكرة كاش ووحدات حماية وتنظيم الذاكرة MPU و MMU وذلك لتنظيم النفاذ للذاكر الداخلية والخارجية (Internal & External). كما يتميز بالمعالج المشترك منخفض الطاقة للغاية (ULP) هناك أيضاً 448 كيلوبايت من ROM و 520 كيلوبايت من SRAM و 4 ميغابايت من ذاكرة فلاش (لتخزين البرامج والبيانات) وهو ما يكفي لمواكبة السلاسل الكبيرة التي تشكل صفحات الويب وبيانات JSON / XML وكل ما نجد على أجهزة إنترنت الأشياء في الوقت الحاضر.

تدمج ESP32 جهاز الإرسال والاستقبال اللاسلكي HT40 من 802.11 b / g / n ، لذلك لا يمكن الاتصال بشبكة Wi-Fi والتفاعل مع الإنترنت فحسب ، بل يمكنه أيضاً إعداد شبكة خاصة به ، مما يسمح للأجهزة الأخرى بالاتصال مباشرةً بها.

تدعم ESP32 أيضًا خدمة Wi-Fi Direct ، وهو خيار جيد للاتصال من نظير إلى نظير دون الحاجة إلى نقطة وصول . تعد Wi-Fi Direct أسهل في الإعداد وأسرع في نقل البيانات من تقنية Bluetooth . تحتوي الشريحة أيضًا على إمكانيات Bluetooth ثنائية الوضع (Dual) ، مما يعني أنها تدعم كل من Bluetooth 4.0 (BLE / Smart) و Bluetooth Classic (BT) ، مما يجعلها أكثر تنوعًا.

بعض الخصائص الكهربائية:

- 1- تردد التشغيل: 2.4 غيغا هرتز
- 2- الحد الأقصى لمعدل البيانات: 150 ميغا بايت / ثانية
- 3- انتاج الطاقة: 20.5 ديسيبل ميلي واط
- 4- الحساسية: -98 ديسيبل ميلي واط
- 5- جهد إمداد التشغيل: 2.3 فولت إلى 3.6 فولت
- 6- تيار الإمداد المستقل : 95 مللي أمبير إلى 100 مللي أمبير
- 7- تيار الإمداد المرسل: 240 مللي أمبير

هذه الشريحة لا تحوي طرفيات فقط بل أيضاً حساسات داخلية وهي:

1. حساس حرارة بمجال من -40 °C – 125 °C.
2. عشرة أرجل قابلة للعمل كحساسات سعوية (حساس لمس) capacitive-sensing.
3. حساس تأثير هول وبهذا؛ باستخدام مغناطيس ودارة ESP32 يمكن صنع تطبيق تحسس لحالة

الباب.

لتعداد بعض الطرفيات الأخرى المتاحة بشكل مختصر:

- 34 دخل/خرج.
- 18 قناة للتحويل التماثلي الرقمي ADC بدقة 12 بت.
- قناتين للتحويل الرقمي التماثلي DAC بدقة 8 بت.
- واجهة إيثرنت Ethernet MAC.
- Ethernet MAC interface with dedicated DMA and IEEE 1588 Precision Time Protocol support
- SD/SDIO/CE-ATA/MMC/eMMC host controller
- SDIO/SPI Slave Controller
- ثلاث وجهات تسلسلية UART.
- واجهتين I²C.
- واجهتين I²S.
- ثمانية أقنية لإرسال واستقبال الإشارة IR.
- جهاز تحكم عن بعد بالأشعة تحت الحمراء (TX / RX ، ما يصل إلى 8 قنوات)
- عداد نبضات
- Pulse Width Modulation (PWM) Motor & LED
- متحكم النفاذ المباشر للذاكرة
- مجموعتين من العدادات 2×64 -بت

ولا ننسى ذكر أهم ميزات الشريحة عن غيرها من ناحية الحماية وهي:

- ميزات الأمان القياسية IEEE 802.11 كلها مدعومة ، متضمنةً WFA و WPA/WPA2

و WAPI

- الإقلاع الآمن

- تشفير محتويات ذاكرة الفلاش

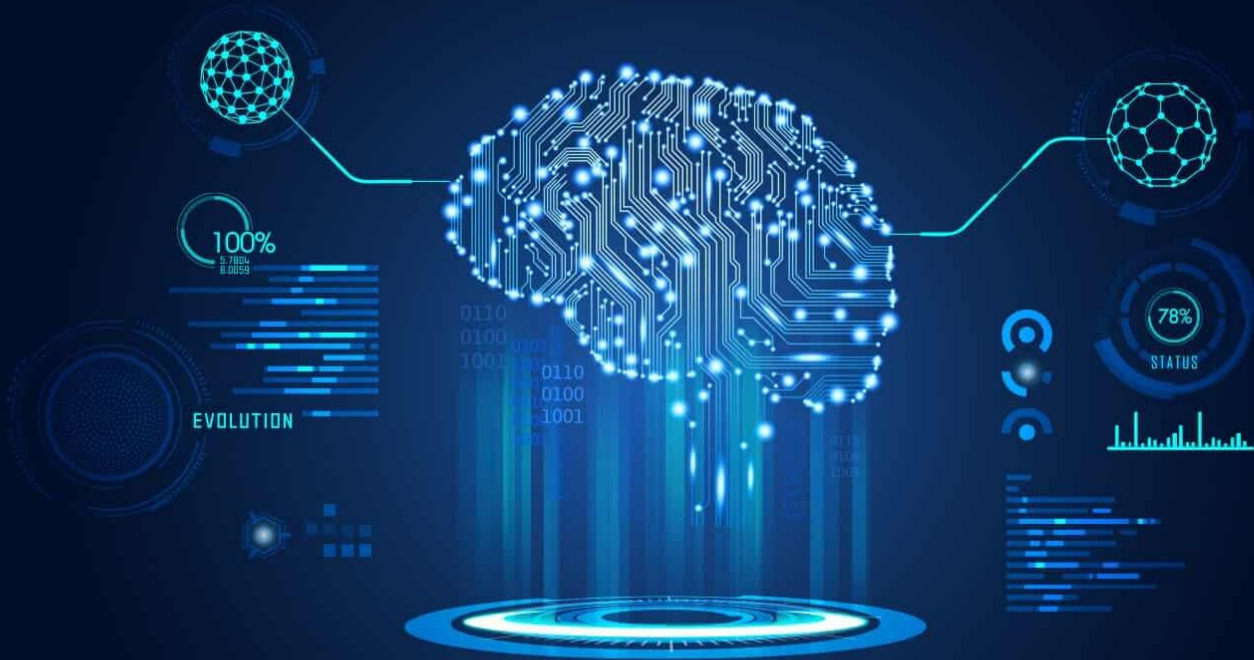
- كلمة سر لمرة واحدة OTP مؤلفة من 1024 بت، ولحد 768 بت للمستهلكين.

- تسريع أجهزة التشفير: AES ، SHA-2 ، RSA ، مشفر المنحنى الإهليجي (ECC) ، مولد

الأرقام العشوائية (RNG)

نهايةً، إن مصممي ESP32 قد أبدوا انتباهاً جيداً لموضوع توفير الطاقة والذي شمل تضمين متحكم خاص للعمل في نمط التوفير الأقصى ultra-low-power processor وتمت تسميته باسم ULP Co-processor ويحوي ذاكرة رام بحجم 8 كيلو بايت للبيانات والتعليمات معاً. هذا المعالج المساعد له بيئته التطويرية الخاصة من ناحية الأدوات والـtoolchain.

الفصل الثالث التعلم الآلي *Machine learning*



القسم الثالث - التعلم الآلي:

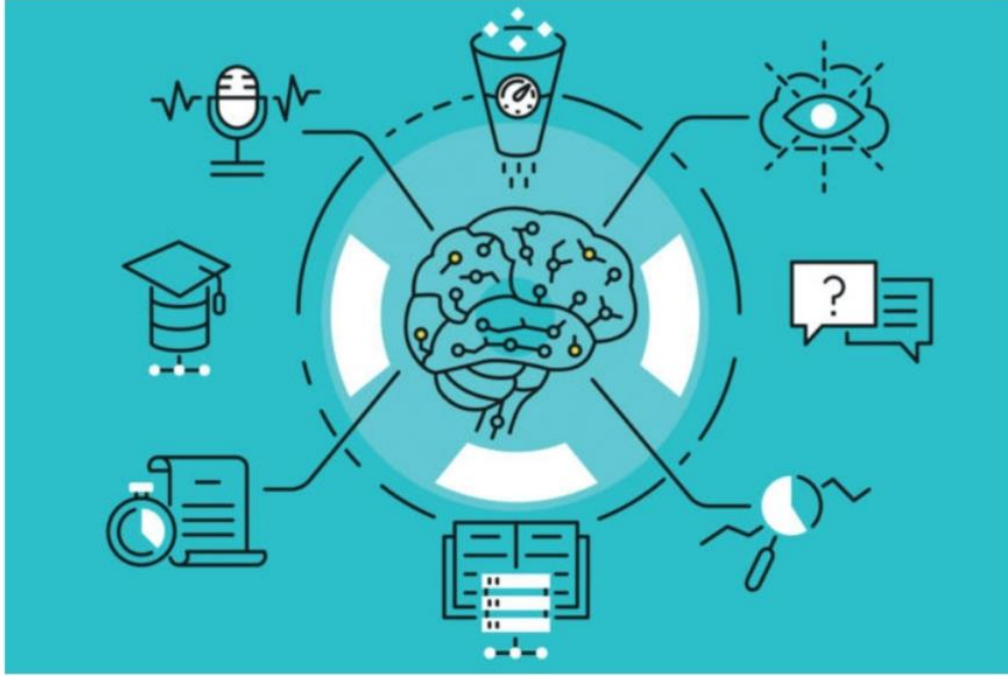
تعلم الآلة أو ما يعرف باسم *Machine learning* وهو واحد من أشهر فروع الذكاء الاصطناعي والذي يعرف بـ *Artificial Intelligence* ويرمز اختصاراً له "AI"، وغالباً ما تتم مناقشة التعلم الآلي والذكاء الاصطناعي معاً، ويتم استخدام المصطلحات في بعض الأحيان بالتبادل، ولكنهما لا يمثلان الشيء نفسه، ومن المهم هنا أن نذكر أنه على الرغم من أن جميع تسميات التعلم الآلي تعد ذكاءً اصطناعياً، ولكن ليس كل ذكاء اصطناعي يمثل تعلماً آلياً.

ومن الممكن أن نعرفه أيضاً بأنه إعطاء أجهزة الكمبيوتر القدرة على التعلم بدون برمجة صريحة لها، ويتم ذلك عن طريق إعطاء البيانات لأجهزة الكمبيوتر ومن ثم تحول البيانات إلى نماذج قرار يتم استخدامها بعد ذلك للتنبؤات المستقبلية.

- تم استخدام مصطلح "التعلم الآلي" لأول مرة في عام 1959 من قبل آرثر صموئيل، وهو أمريكي عمل في شركة آي بي إم *IBM* في الستينيات، حيث كان التعلم الآلي يستخدم في الغالب لتصنيف وتمييز الأنماط.

وقد سمي بتعليم الآلة لأنه يعتمد على آلة يتم تصميمها لتحاكي العقل البشري في تصرفها، حيث يُسمح للحواسيب أو الآلات بامتلاك خاصيتي "التعلم والتمثيل المنطقي للمعرفة والتفكير" وذلك انطلاقاً من تصميم برامج لديها القدرة على التعلم من الخبرات والمعطيات السابقة عن طريق رسم وتطوير العديد من الخوارزميات والتقنيات، مما يمكننا من إنشاء نموذج حقيقي انطلاقاً من معلومات موجودة سواء قمنا بتحسين نموذج موجود أو قمنا بإنشائه من جديد.

وبإمكاننا إن نختصر الغرض الأساسي من التعلم الآلي بأنه استنتاج معلومات جديدة من بيانات تدريبية سابقة، حيث يتم الاستفادة من هذه المعلومات في بناء نماذج قادرة على التنبؤ بشكل البيانات المستقبلية.



ويوجد هناك مستويان من التعلم الآلي :

1. التعلم الاستقرائي

2. التعلم الاستنتاجي

التعلم الاستقرائي : حيث يقوم الاستقرائي باستنتاج قواعد وأحكام عامة من بيانات ضخمة.

التعلم الاستنتاجي: يعتمد على الانطلاق من أحكام عامة ويطبقها في أمثلة خاصة.

ويستنتج مما سبق بأن المهمة الأساسية للتعلم الآلي هي استخراج معلومات ذات قيمة من البيانات، وبالتالي

هو قريب جداً من التنقيب في البيانات *Data mining*.

ويرتبط التعلم الآلي ارتباطاً وثيقاً بالإحصائيات الحسابية التي تركز على عمل التنبؤات باستخدام أجهزة الكمبيوتر.

بعض أنواع خوارزميات التعلم:

• التعلم تحت المراقبة والإشراف:

يكون في هذا النوع علاقة رياضية تحكم العلاقة التي ما بين الدخل و الخرج ولكنه ليس من الضروري ان تعلم عند استخدامه أي نوع من العلاقات الرياضية تلك، ولكن ما يهم هو بأن الخرج سيكون معروف بالنسبة لنا.

فمثلاً إذا أردنا أن نتنبأ بفترة إلغاء المستخدمين اشتراكاتهم في أي خدمة ما قد تكون مقدمة لهم، فهنا ما نعرفه في نتيجة استخدام هذا النوع هو "هل سيقوم المستخدم x بإلغاء اشتراكه أم لا" و لكن ليس من الضروري معرفة "كيف قام بمعرفة فيما إذا كان سيقوم بإلغاء أو لا ..".

وليقوم هذا النوع بمهمته يتطلب تخصيص جزء من البيانات لـ "التعلم" وجزء من البيانات للتحقق من مدى دقة النموذج و قياسه أي يقوم بـ "الاختبار"

وسمي بالإشراف لأننا نشرف على إعطائه المخرجات للبيانات، ومن خلال التدريب يبني الحاسوب علاقات وأنماط بين البيانات و المخرجات ليستطيع لاحقاً من التنبؤ بمخرجات بيانات جديدة.

ويوجد هناك نوعان من التعلم الخاضع للإشراف هما:

1. **التصنيف Classification**: وهو التنبؤ بتصنيفات مثل أنواع الحيوانات او النباتات.
2. **الانحدار أو التوقع Regression**: وهو التنبؤ بالكمية، فمثل توقع درجة حرارة اعتماداً على بيانات تاريخية، وبالإضافة إلى اتجاه الرياح و الغيوم.

• التعلم غير خاضع للإشراف:

بهذا النوع نحن لا نعرف ما نريد استخراج من النموذج، ولكن نعتقد بأن هناك بعض الأنواع العلاقات أو الارتباط *correlation* بين البيانات المتوفرة لدينا، وقد تكون البيانات معقدة للغاية و لا يمكن التكهّن بها، لذا نقوم بتنظيم البيانات المتوفرة *normalize* في صيغة منطقية للمقارنة، ومن ثم ترك النموذج يعمل عليها و محاولة العثور على بعض العلاقات التي تربط بين البيانات. وتتمثل إحدى الخصائص المميزة لهذه النماذج بأنها تقوم باقتراح طرق مختلفة لتصنيف البيانات و تنظيمها.

فعلى سبيل المثال: بعد معالجة جميع البيانات المتعلقة بجميع مستخدمين منتج معين باستخدام خوارزمية غير خاضعة للإشراف، فقد يتم إنشاء طريقة لتقسيم المستخدمين إلى مجموعتين، وبعد

فحص ومقارنة هاتين المجموعتين، فقد نجد أن المجموعة (أ) في موقع جغرافي معين، والمجموعة (ب) في موقع آخر، ويمكننا التصرف بناءً على هذا التقسيم المحدد للبيانات، وإذا لم يكن هناك نتيجة من هذا التقسيم مستفاد منها يمكننا إعادة ترتيب البيانات و الحصول على نتائج مختلفة. ففي التعلم الغير خاضع للإشراف يحاول النظام اكتشاف البنية الخفية للبيانات أو الارتباطات بين المتغيرات، وفيه يجمع البرنامج البيانات المتشابهة إلى مجموعات، ومن ثم يتم تصنيف عينة الاختبار بناءً على قربها أو بعدها من هذه المجموعات. و من أشهر الخوارزميات التعلم الغير خاضع للإشراف خوارزمية *K-Means*، ومن التطبيقات التي تستخدم هذا النوع من التعلم كثيراً:

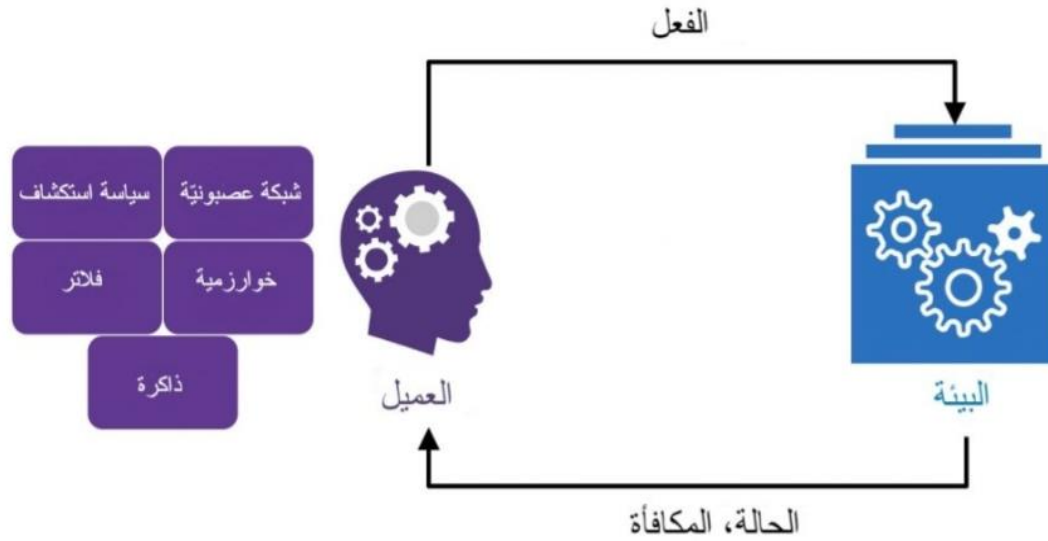
تجميع العملاء أصحاب التفضيلات المتشابهة في نفس المجموعات أو اكتشاف و تصنيف المجتمعات (الأشخاص ذوو الاهتمامات المشتركة) في وسائل التواصل الاجتماعي.

● تعلم المعزز *Reinforcement Learning* :

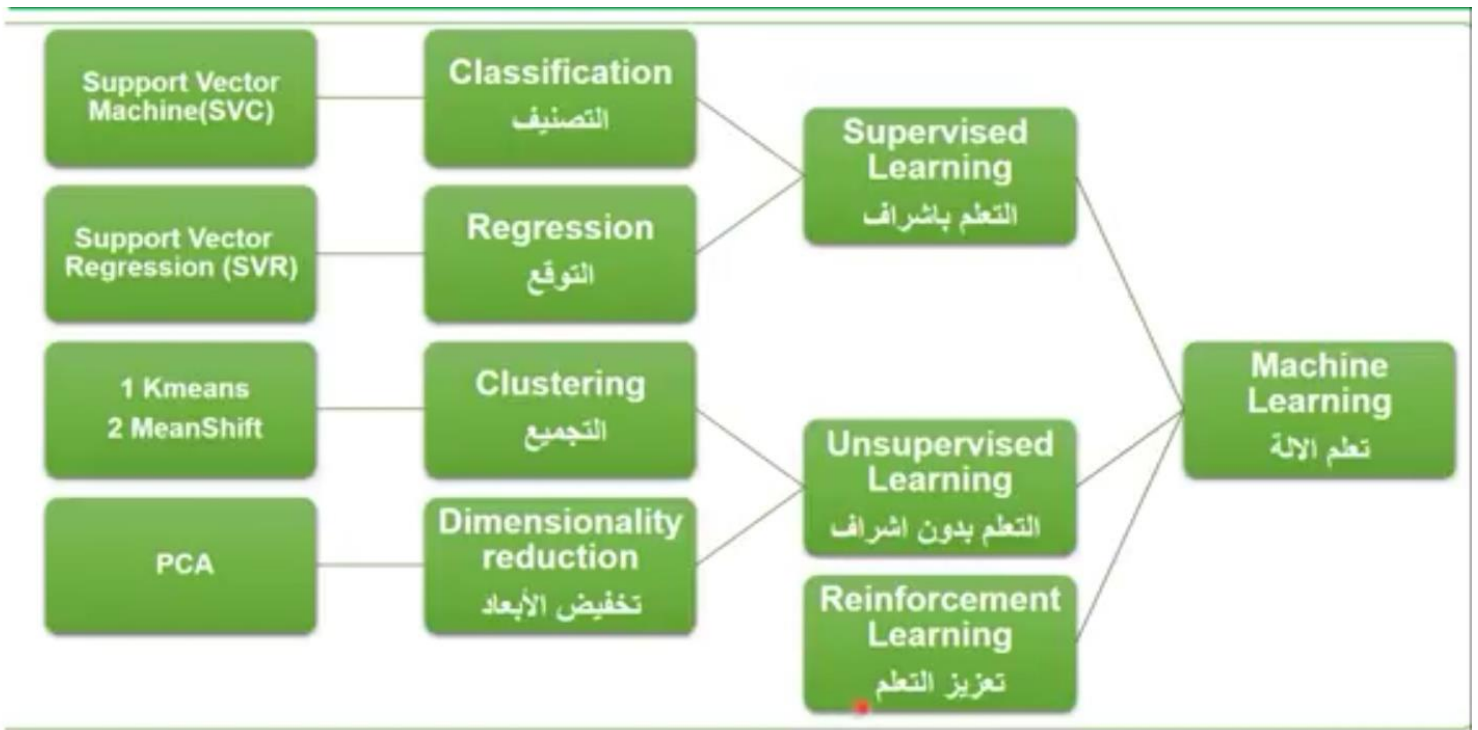
هو أحد أهم مجالات التعلم الآلي حالياً، ويتلخص بأن العميل *agent* (مثلا البرنامج) يتبادل التأثير مع البيئة المحيطة وعلى أساس تفاعله مع هذه البيئة يقوم بالتعلم بحيث يؤدي الأفعال التي تعطيه المكافأة الأكبر.

أي تشبه هذه الطريقة طريقة تعلم الإنسان حيث إنه عندما يريد التعلم أي شيء قد يرتكب أخطاء و يتعلم منها و تزيد خبرته و هذا هو الحال أيضا في هذه الخوارزمية المتبعة. أي أن التعليم المعزز هي عملية تستند إلى تجربة لا إلى البيانات التدريب *training data*، أي إنه يجب على العميل أن يقوم بمحاولات عديدة و استكشاف بيئته من أجل الحصول على مكافآت ثم تحديث السياسة المتبعة تبعاً لذلك.

توضح هذه الصورة الآلية المتبعة في التعليم المعزز:



وهنا نستعرض أهم الخوارزميات المتبعة في كل نوع من الأنواع الثلاثة:



• ونعرف أهم الخطوات العامة للتعلم الآلي :

1. تجميع البيانات
2. تصفية البيانات
3. تحليل البيانات
4. تدريب الخوارزمية
5. اختبار الخوارزمية

خوارزمية آلة المتجه الداعم *Support vector machine* :

إتقان خوارزميات التعلم الآلي و استخدامها بشكل يتلاءم مع طبيعة المشكلة التي لدينا، ليست خرافة على الإطلاق ، فإمكاننا إن نمثل هذه الخوارزميات وكأنها مستودع معبأ بالفؤوس و السيوف وبعض الأدوات الأخرى ،ولكن يتوجب علينا أن نتعلم كيفية استخدامها في الوقت المناسب فمثلاً الانحدار يشبه السيف ،حيث إنه قادر على تشريح البيانات وتقسيمها بكفاءة عالية نوعاً ما ،و لكنه غير قادر على التعامل مع البيانات المعقدة للغاية ،وعلى العكس من ذلك الـ *SVM* يشبه السكين الحاد فهو يعمل مع حجم بيانات أصغر ،و أحياناً المعقدة منها.

وهذا ما جعل خوارزمية التعلم آلة المتجه الداعم من اهم الخوارزميات المستخدمة بكثرة في السنوات الأخيرة ،و تستعمل بشكل مخصوص في مهام التصنيف و الانحدار ،وتعتمد على فكرة تحويل البيانات الأصلية إلى نمط جديد حيث يمكننا من خلاله فصل البيانات و تصنيفها.

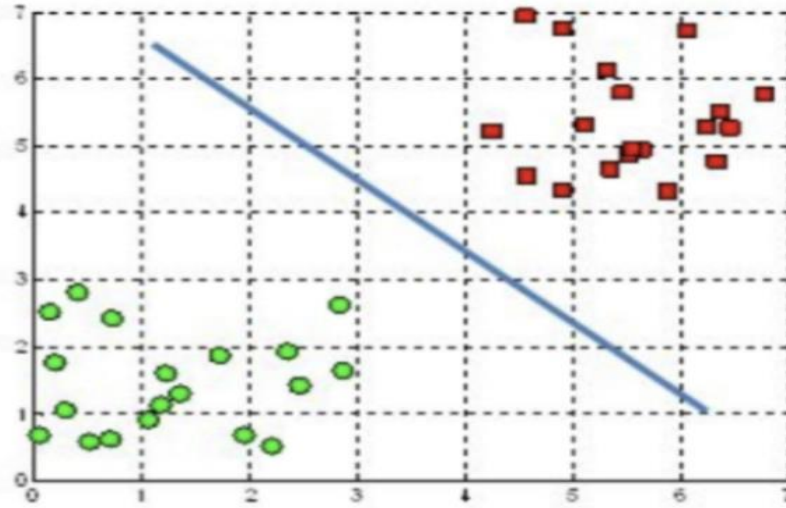
و بإمكاننا أن نقول عنها بإنها عبارة عن نظرية تجمع بين النظرية الإحصائية و التعلم بإشراف ،و التي تم تطويرها من قبل الباحث *Vapnik* في عام 1998.

تعتمد فكرة خوارزمية آلة المتجه الداعم على البحث على أفضل طريقة لتقسيم البيانات إلى مجموعتين، وذلك بوضع مستوي فائق *hyperplane* بينهما وذلك بغض النظر عن طبيعة البيانات سواء أكانت قابلة للفصل الخطي أو لا، وهنا يكمن مصدر قوتها.

وفيما يلي سنوضح بعد المفاهيم التي ذكرت أعلاه ...

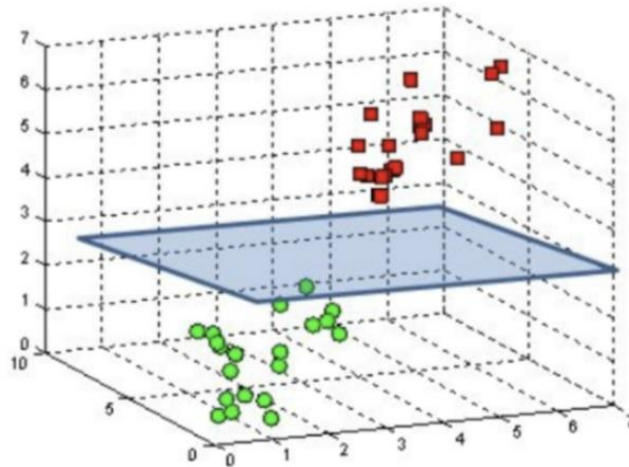
المستوي الفائق *Hyperplan* :

ويقصد به الأداة التي يمكن بها فصل البيانات و يختلف حسب الفضاء الممثلة بها البيانات، فإذا كان الفضاء ثنائي البعد في حال المزايا المدخلة عددها (2) فسيكون خطاً مستقيماً.. كما يظهر في الشكل:



الشكل 3: المستوي الفائق في فضاء ثنائي البعد.

أما في حال كان عدد المزايا المدخلة (3) فسيكون المستوي الفائق على شكل مستوى كما يوضحه الشكل التالي:

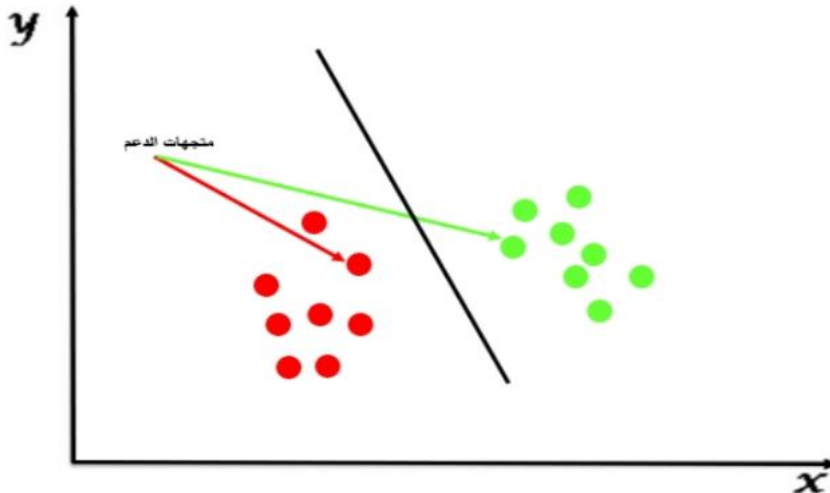


الشكل 4: المستوي الفائق في فضاء ثلاثي البعد.

وفي حالة الفضاء المكون من n بُعد يكون عدد المزايا المدخلة n .

متجهات الدعم : *Support Vectors*

تعتبر المتجهات الدعم هي أقرب نقاط البيانات الأقرب على المستوى الفائق، وهي بمثابة نقاط ذات أهمية كبيرة في عملية تصنيف البيانات لأن من خلالها يتم تحديد المستوى الفائق الأفضل في عملية فصل البيانات، وبالتالي إزالتها أو تغيير موقعها يتطلب تحديد مستوى فائق آخر من جديد.

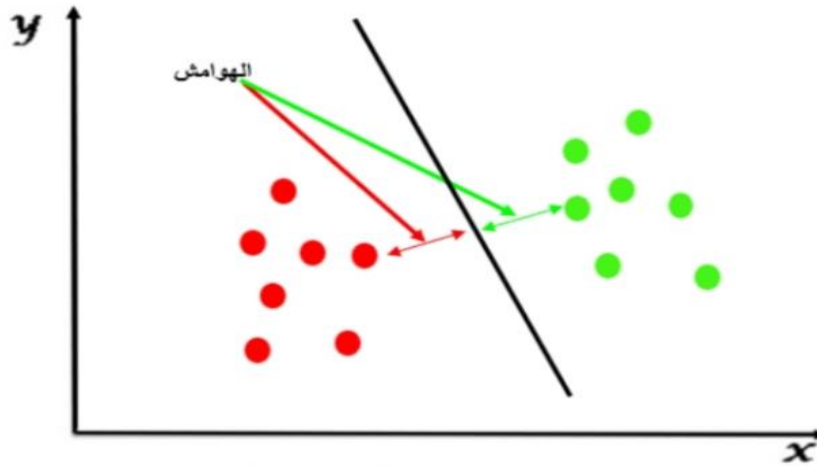


الشكل 5: متجهات الدعم.

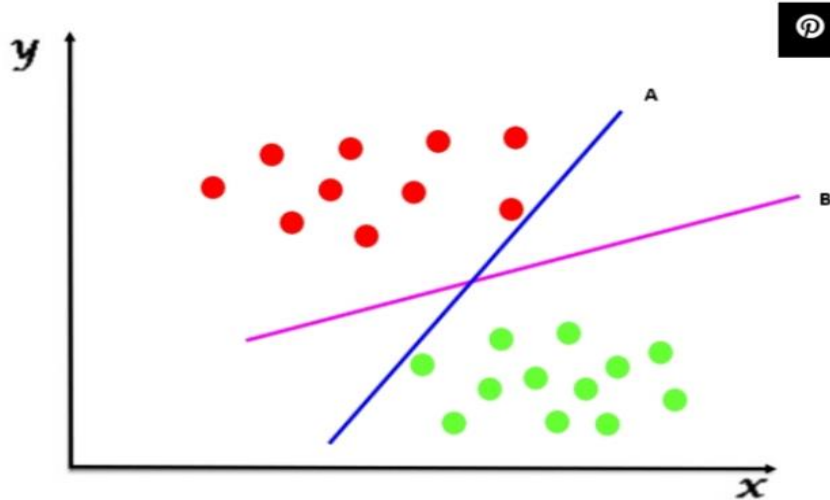
الهامش : *Margin*

هو عبارة عن المستوى الفائق و أقرب نقطة من نقاط مجموعة البيانات كما يوضح الشكل (6):

بحيث كلما كانت هذه المسافة كبيرة كلما زادت احتمالية تصنيف البيانات الجديدة بشكل صحيح كما هو واضح في الشكل (7) فنلاحظ أن المستوى الفائق B هو أفضل من المستوى الفائق A ، وذلك لأن المسافة بين نقاط المتجهات الدعم أكبر ما يمكن ، وهذا ما يتيح إمكانية تصنيف أكبر عدد من النقاط بشكل صحيح ، وبالتالي تكمن أهمية الهامش في الحصول على الموضع الأفضل للمستوى الفائق.



الشكل 6 : الهوامش.



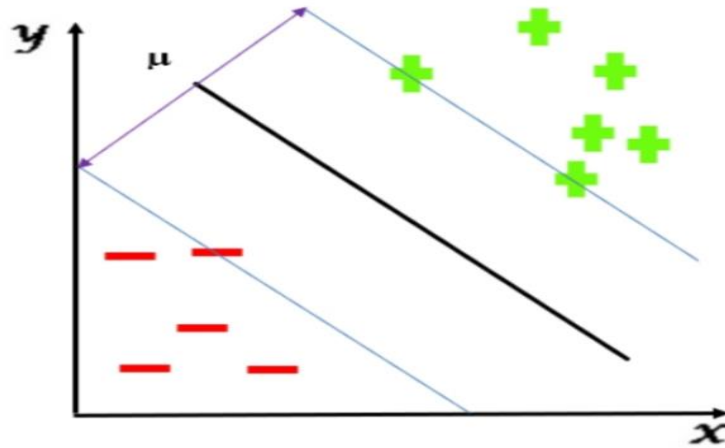
الشكل 7 : تقييم واختيار المستوى الفائق الأفضل.

الفرق بين الهامش اللين *Soft margin* و الهامش الصلب *Hard margin*:

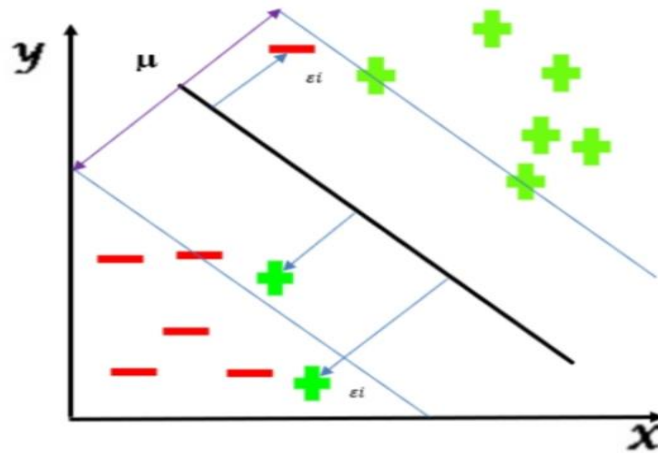
في الهامش الصلب تكون مجموعة البيانات قابلة للفصل خطياً أي النقاط غير متداخلة مع بعضها البعض كما في الشكل (8)

أما في الهامش المرن تحتوي مجموعة البيانات على بعض النقاط القليلة التي تمنع عملية الفصل الخطي، و في هذه الحالة يتم السماح لهذه النقاط أن يتم تصنيفها بشكل خاطئ سواء أكانت ضمن منطقة الهامش أو

خارجة، و بالتالي يتم إضافة معامل يعبر عن نسبة الخطأ في كل نقطة من نقاط التدريب سواء أكانت مصنفة بشكل صحيح أو خاطئ.



الشكل 8: هامش صلب.

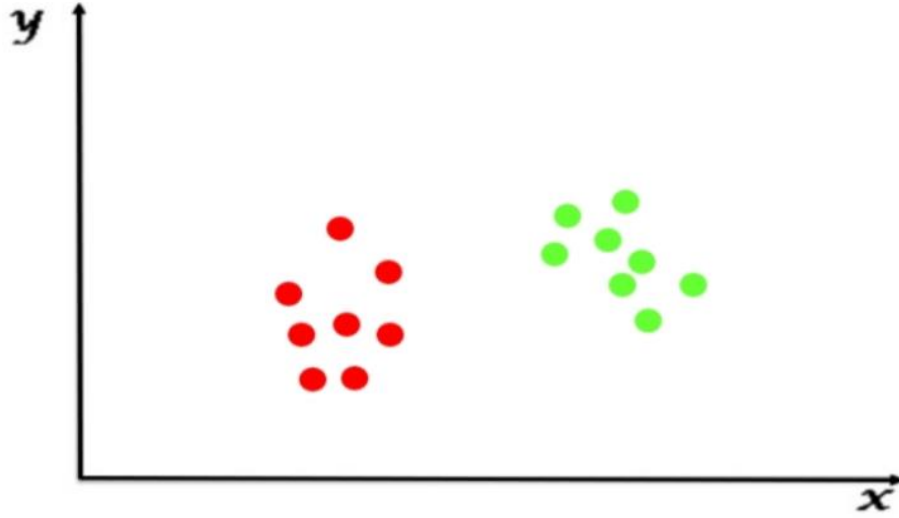


الشكل 9 : هامش مرن.

أنواع متجهات دعم الآلة :

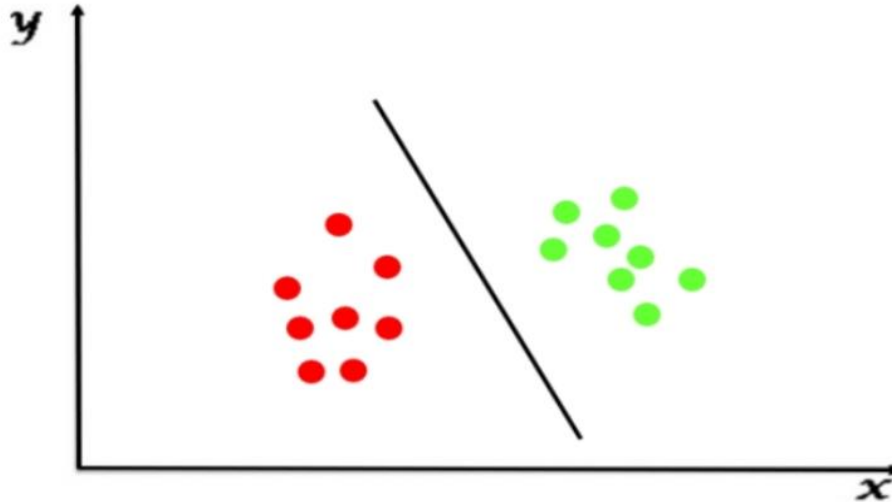
آلة المتجه الداعم الخطي *Linear SVM*:

عبارة عن مصنف *classifier* يتم استخدامه لفصل البيانات التي تتميز أنها قابلة للفصل خطيا ، كما في الشكل (10) بواسطة مستوي فائق يعبر عن مستقيم مهمته فصل مجموعة البيانات إلى مجموعتين.



الشكل 10: مجموعة بيانات قابلة للفصل الخطي.

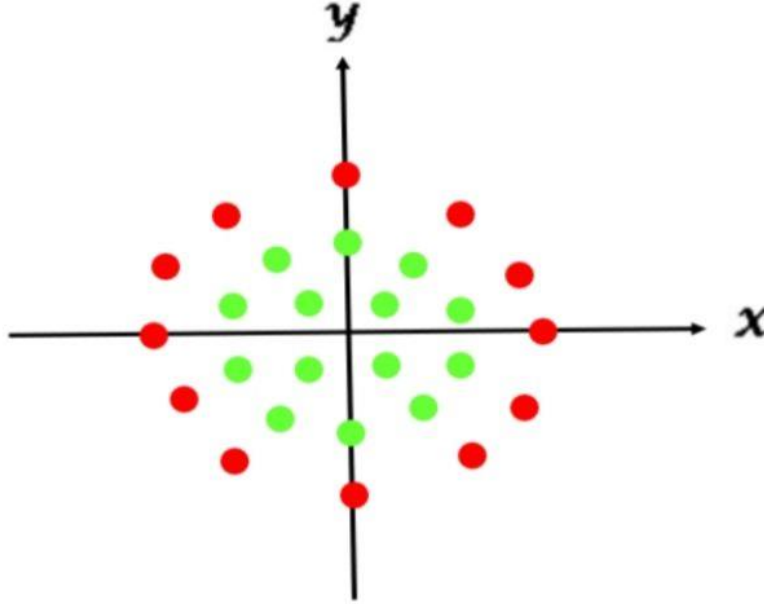
يدعى هذا المصنف بـ *Linear SVM classifier* أي مصنف آلة متجه الداعم الخطي، ومبدأ عمله يعتمد على إيجاد القيم المثالية لمعادلة الخط المستقيم التي تفصل هذه البيانات بأفضل شكل ممكن كما في الشكل (11) أي بأكبر مسافة للهامش ما بين أي نقطة من نقاط المتجهات الدعم و المستوى الفائق .



الشكل 11: فصل مجموعة البيانات بواسطة خط مستقيم.

آلة المتجه الداعم غير الخطي *Non linear SVM*:

عبارة عن مصنف يتم استخدامه لفصل البيانات التي تتميز بأنها غير قابلة للفصل خطيا بواسطة مستقيم كما في الشكل (12) ويدعى هذا المصنف بـ *Non linear SVM classifier* أي مصنف آلة المتجه الداعم غير الخطي.



الشكل 12 : مجموعة بيانات غير قابلة للفصل الخطي.

و مبدأ عمله ينص على تحويل مجموعة البيانات التي لدينا من فضاءها الحالي أي من الفضاء الثنائي البعد إلى فضاء أعلى ، حيث يمكننا فصل البيانات و تمييزها، وبالتالي تبدأ الخوارزمية بشكل آلي بتجريب إضافة بعد ثالث للبيانات .

ولیکن كاحتمال أول Z وبعدها تُختبر مجموعة البيانات الممثلة في فضاء ثلاثي الأبعاد X, Y, Z هل أصبحت قابلة للفصل أم لا ؟!

إذا تحقق الشرط عندها تبحث الخوارزمية على المستوي الفائق المناسب أكثر لعملية الفصل ربما يكون

مستوي ثلاثي الأبعاد أو مُجسم معين ، و إذا لم يتحقق الشرط يتم إضافة بُعد رابع و لیکن M و بعدها تختبر

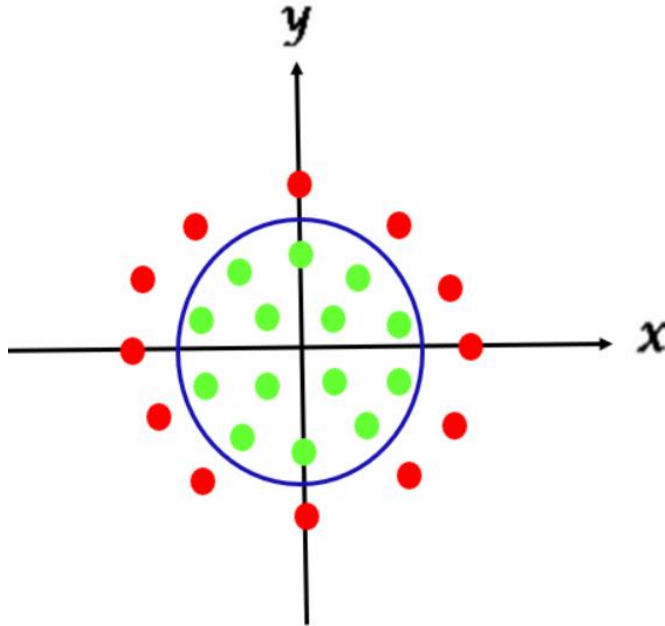
مجموعة البيانات الممثلة في فضاء رباعي الأبعاد X, Y, Z, M هل أصبحت قابلة للفصل أم لا و هكذا...

تستمر الخوارزمية بإضافة أبعاد حتى تصل لمرحلة تُصبح بها البيانات قابلة لفصل و هذه الطريقة تسمى طريقة النواة *Kernel method* .

نبدأ بتجريب احتمال إضافة البعد الثالث ،وتكون معادلته الأنسب لفصل البيانات هي دائرة و بالتالي تكون معادلة البعد الثالث كالتالي:

$$Z = X^2 + Y^2 \quad (1)$$

وعندما تكون $Z=1$ تمثل معادلة دائرة نصف قطرها واحد، وبالتالي إذا قمنا بتحويل مجموعة البيانات من فضاء ثلاثي الأبعاد إلى فضاء ثنائي البعد أي التخلص من البعد الثالث Z من خلال إعطائه قيمة واحد سنحصل على دائرة جميع نقاط البيانات المحتواة داخلها تنتمي إلى صنف النقاط الخضراء ، والنقاط المتبقية الواقعة خارجها تنتمي إلى صنف النقاط الحمراء كما الشكل التالي:

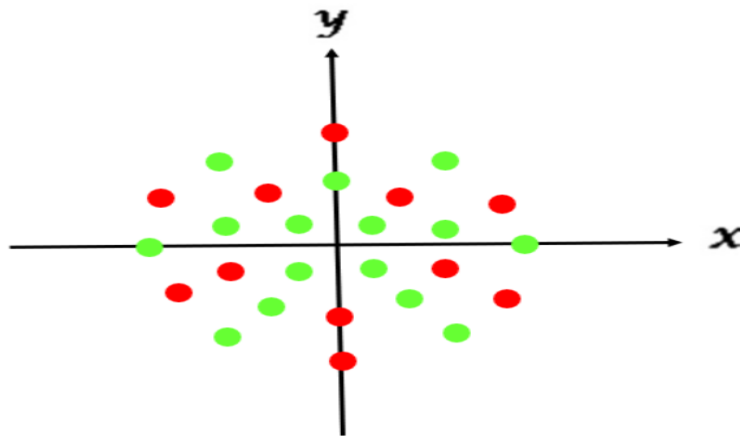


إيجابيات وسلبيات خوارزمية آلة المتجه الداعم SVM الإيجابيات:

- تعمل بشكل فعال مع البيانات القابلة للفصل الخطي أي التي تملك هامش فصل واضح.
- تُعطي نتائج أكثر دقة في حال كان عدد مزايا بيانات التدريب كبيرة.
- تُعطي نتائج أكثر دقة في حال كان عدد مزايا بيانات التدريب أكبر من عدد أمثلة التدريب.
- كفاءتها عالية لأنها تستخدم جزء من بيانات التدريب في اتخاذ القرار ويقصد بذلك يتم اختيار المستوى الفائق الأفضل اعتماداً على نقاط البيانات الواقعة على المتجهات الداعمة فقط وليس اعتماداً على كامل بيانات التدريب.

السلبيات:

- لا تعمل بشكل جيد للغاية عندما يكون لدينا مجموعة بيانات كبيرة لأن المدة الزمنية اللازمة لتدريبها عالية.
- لا تعمل بشكل جيد للغاية عندما تكون مجموعة البيانات تحوي أخطاء كثيرة , سواء كانت ضجيج $noise$ أثناء جمع البيانات أو متداخلة بشكل كامل كما في الشكل:



الرؤية الحاسوبية:

هي إحدى مجالات علم الحاسوب التي تتضمن طرق تحصيل و معالجة و تحليل وفهم الصور، وتهدف الرؤية الحاسوبية إلى بناء تطبيقات قادرة على فهم محتوى الصور كما يفهمها الإنسان بهدف الاستفادة منها في تطبيقات مختلفة و تعتبر أيضاً إحدى المجالات التي أحدثت ثورة في التطبيقات الهندسية الحديثة الروبوتية منها و الصناعية.

يمكن للصور الرقمية أن تأخذ عدة أشكال مثل صور متتابعة (فيديو)، مشاهدة من عدة كاميرات أو صورة متعددة الأبعاد من الماسحات الطبية.

تطبيقات الرؤية الحاسوبية :

1-تطبيقات الواقع الافتراضي.

2- الاستخدامات الطبية وتحليل الصور الطبية و تشخيص بعض الأمراض.

3-عمليات التحكم الصناعية و الروبوتات الصناعية.

4-عمليات الملاحة كما في العربات ذاتية الحركة والروبوت النقال.

5-كشف الاحداث والأشخاص مثل المراقبة المرئية والتعرف على الأشخاص.

6-تصنيف المعلومات والمعطيات وفهرسة قواعد البيانات للصور والفيديو.

واحدة من أهم تطبيقات الرؤية الحاسوبية هي الرؤية الحاسوبية الطبية او معالجة الصور الطبية وهذا المجال يعتمد على استخلاص المعلومات من الصور الرقمية بغية الوصول إلى تشخيص طبي للمريض ،وبشكل عام تكون المعلومات على شكل صور مايكروسكوبية أو صور شعاعية أو صور للأوعية أو صور فوق صوتية.

وكما يعد من أحد تطبيقاتها استخدامها في الصناعة، وتدعى برؤية الآلة حيث تكون المعلومات المستخلصة من الصورة تهدف إلى تطوير العملية الإنتاجية مثلاً تحديد مكان وإحداثيات واتجاه القطع لكي تقوم ذراع روبوت بالتقاطها.

وكتطبيق آخر للرؤية الحاسوبية في الصناعة ،عملية التحكم في الجودة حيث يتم فحص المنتج النهائي بشكل آلي للتأكد من عدم وجود أية أخطاء او عيوب إنتاجية.

وكما تستخدم أيضا في التعرف على الوجوه كشف الهويات للأشخاص، بحيث تستخدم هذه التقنية مثلا كتعرف على الوجه كأحد أساليب الحماية.

المهام الرئيسية للرؤية الحاسوبية:

• التعرف *Recognition*:

هي المهمة التقليدية في الرؤية الحاسوبية ،وهي القيام بتحديد ما إذا كانت الصورة تحتوي أو لا تحتوي جسماً معيناً.

هذه المهمة من الممكن حلها ببساطة وبدون جهد يذكر بواسطة الإنسان لكن لا تزال هذه المسألة غير محلولة بشكل فعال ونهائي من قبل الحاسوب في شكلها العام.

جميع الطرق الموجودة لحل هذه المسألة تقوم بإيجاد أفضل الحلول من أجل إيجاد أشكال معينة كالأشكال الهندسية، وجوه الأشخاص، الأحرف المطبوعة أو المكتوبة أو السيارات ،وفي حالات معينة فقط محددة على الغالب بظروف إضاءة محددة، خلفية ووضعية معينة للجسم بالنسبة للكاميرا.

• التحديد *Identification*:

تحديد مطابق وحيد للجسم المعرف فمثلاً: تحديد وجه شخص معين أو التعرف على بصمة شخص معين أو سيارة من نوع معين.

• التحري *Detection*:

يتم البحث في بيانات الصورة لإيجاد جسم معين.

مثال: تحري وجود خلايا مريضة في صورة طبية أو التحري عن وجود سيارة مسرعة على الطريق أو

كشف شيء مميز بالوجه كتمييز فيما إذا كان يرتدي كمامة أو لا كما هو الحال في مشروعنا..



معالجة الصورة الرقمية *Digital Image processing*:

هو أحد الفروع علم الرؤية الحاسوبية، والتي تهتم بإجراء عمليات على الصور بهدف تحسينها طبقاً لمعايير محددة أو استخلاص بعض المعلومات منها.

حيث يتألف نظام معالجة الصور التقليدي من ستة مراحل متتالية وهي على الترتيب:

1. استحصال الصورة *Image acquisition*

2. المعالجة المبدئية *pre_processing*

3. تقطيع الصورة *segmentation*

4. استخلاص المميزات *feature extraction*

5. تصنيف المميزات *classification*

6. فهم الصورة *image understanding*

حيث يتم استحصال الصورة بوساطة حساس ضوئي (كاميرا) ومن ثم يتم تصفية الصورة من التشويش وتحويلها إلى صورة ثنائية و ثم تفصل المعلومات الهامة في الصورة عن الخلفية وتحدد الصفات الهامة والمميزة في الصورة وربطها بالنمط الذي تعود إليه والتعرف على الأنماط. وتستخدم نظم معالجة الصورة في الكثير من التطبيقات ولا سيما تطبيقات الرؤية الحاسوبية في التحكم الآلي والروبوتات.

أساسيات معالجة الصورة *Image Processing*

الصورة من وجهة نظر الحاسب الآلي:

الصورة هي عبارة عن مصفوفة أبعادها تمثل أبعاد الصورة الحقيقية (بالبكسل) فإذا كانت الصورة 600×480 بيكسل إذا المصفوفة أبعادها 600×480 .

البكسل:

البكسل هو جزء من الصورة ويمثل مربع من مربعات الصورة ويحتوي على قيمة معينة تبعا للون الذي يحتويه هذا البكسل أو هذا المربع من الصورة.

مفهوم الكاميرا 8 غيغا بيكسل:

أي أن مساحة الصورة التي تلتقطها تساوي 8 غيغا بيكسل.

مثال: إذا كان هنالك كاميرا تلتقط صورة بحجم 1536×2048 إذا الناتج سيكون 3.2 مليون بيكسل أي 3.2 ميغا ، وذلك بحيث كلما زاد عدد البكسل في الصورة كلما كانت الصورة أوضح وأبقى.

آلية قراءة الحاسب للصورة:

ذكرنا سابقا ان الصورة في الحاسب تمثل المصفوفة أبعادها تساوي أبعاد الصورة وعناصرها تساوي عدد البكسل ... وكل عنصر يحتوي قيمة اللون الذي تحتويه البكسل.

قيمة اللون في البكسل:

الصورة عندما تلتقط لها أكثر من شكل...فهناك صور أبيض وأسود فقط... وهناك صور رمادية وهي عبارة عن أبيض وأسود... وهناك صور ملونة ...

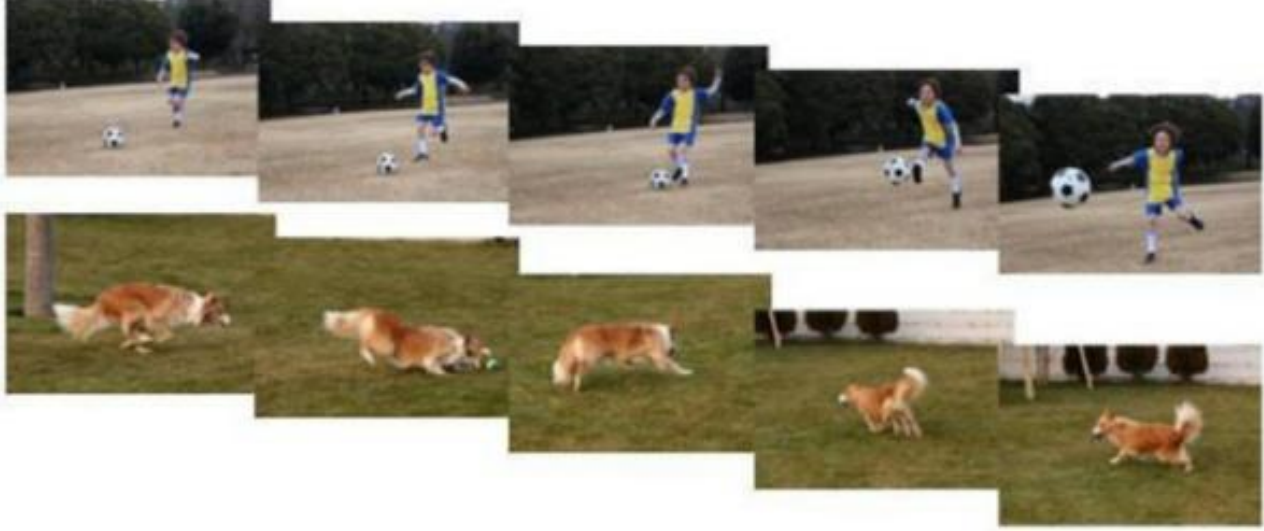
وهنا نأتي لتعرف مهم في *Image Processing* وهو *number of bits per pixel*: أي عدد ال *Bits* التي تمثل فيها البيكسل الواحدة واختصارها *BPP*

فإذا كانت الصورة 1 *BPP* يعني *bit* واحد فقط وهذا يستخدم في حال الصورة أبيض وأسود .

أما في حال الصور الملونة فالكاميرا تلتقط 3 مرات للصورة الواحدة حيث تقيس اللون الأحمر ومرة تقيس اللون الأخضر ومرة تقيس اللون الأزرق ... وتسمى *RGB* أي ان الصورة هي عبارة عن ثلاث مصفوفات حيث مصفوفة للون الأخضر ومصفوفة للون الأحمر ومصفوفة للون الأزرق ،وهنا البيكسل الواحدة تمثل في 24 بت بحيث ثمانية للأحمر وثمانية للأخضر وثمانية للأزرق إذا عدد احتمالات الألوان البيكسل ككل يساوي 2^{24} ويساوي 16777216 لون .

مفهوم الفيديو أو البث الحي:

البث الحي أو الفيديو ما هو إلا عبارة عن النقاط عدد كبير من الصور في الثانية الواحدة وعرضها بالترتيب وتسمى *number of frames per second* وهنا تسمى الصورة بـ *frame* وكلما زاد الفريم في الثانية الواحدة كلما كان الفيديو مزامن للحقيقة.



مفهوم معالجة الصور والعمليات الرياضية التي تجري عليها:

بما أن الصور تمثل في مصفوفات إذا يمكننا إجراء عليها كل العمليات الرياضية التي تجري على المصفوفات: بداية من الجمع والطرح حتى المعادلات والتحويلات (*transformations*) والمقلوب وغيرها...

العمليات الرياضية والغاية منها :

- **الجمع:** إذا جمعت على صورة ما 3 مصفوفات كلها تحتوي على عنصر يساوي واحد فستجمع على مصفوفة الأحمر والأزرق والأخضر... فإذا زادت قيمة الأحمر والأخضر والأزرق فإنك تقترب من الأبيض قليلا فستجد أن الصورة أصبحت أكثر إضاءة.
- **الطرح:** إذا التقطت صورتين لنفس الجسم في نفس الظروف وطرحت الصورتين من بعضهما ستحصل على صورة جديدة تحتوي الفروقات بينهم (بمعنى أنك إذا التقطت صورة لجسم مرة ثم وضعت عليه

جسم آخر والنقط مرة أخرى وطرحت الصورتين ستجد الناتج عبارة عن صورة سوداء لا تحتوي إلا الجسم الذي وضعته مؤخراً...).

- **التحويلات:** كثيراً ما نسمع عن التشويش أو الاهتزاز (البكسلة) وهي عبارة عن صورة تم التقاطها بسرعة فأصبحت الصورة غير واضحة وهنا يتم إجراء عليها تحويلات تقوم بعمل *filter* لتشويش وإعادة الصورة واضحة مثل: (*fourier transformation, geometric transformation*).
- كما وضعنا سابقاً إن كل العمليات على الصور هي مجرد عمليات رياضية بحتة.. ولكن ليس من المعقول إن المبرمجين سيقومون بعمل كل هذه العمليات في كل مرة يستخدمون فيها الصور ولذلك تم إنشاء مكتبة الـ *opencv* التي هي عبارة عن مكتبة تحتوي على دوال تقوم بعمل معظم عمليات الصور التي يحتاجها المبرمجين.

كما تحتوي على دوال تقوم بتحديد أجسام معينة في الصورة مثل الخطوط والأشكال الهندسية ودوال أخرى لتحديد الألوان.

كما يمكن استدعاء ملفات *XML* خاصة بتحديد وجه الإنسان وعينه وشعره وجسمه ككل. ويمكن استخدامها في التعرف على الشخص كما يمكن عمل مصنف *classifier* خاص بأي جسم (النقاط صور عديدة لنفس الجسم من كل الاتجاهات حتى تتعرف عليه الكاميرا من كل الاتجاهات).. وليس هذا مقتصر على الإنسان فقط بل على أي جسم.

خوارزمية فيولا جونز

تجمع هذه النظرية بين خوارزميات ورؤى جديدة معا في إطار واحد، وذلك لأداء مهمة عالية الأداء في الكشف.

ولقد تم عرض هذه النظرية بدافع وبشكل مخصص من أجل كشف الوجه، ولهذا تم إنشاء نظام ليقوم بهذه المهمة وبمعدلات خطأ مقبولة إلى ما حد بعيد.

حيث يتميز هذا النظام بأنه يقوم بكشف الوجه بشكل أوضح من الأساليب الأخرى المتبعة سابقاً كما في نظام *intel Pentium III* الذي يعمل على سرعة 15 إطار في ثانية ويعتمد على الاختلافات الموجودة في الصور في تسلسل الفيديو ليقوم بعملية الكشف أو النظم التي تعمل على إيجاد الاختلافات في ألوان

البكسلات في الصور الملونة، ويحقق أيضا هذا النظام معدلات إطار عالية جدا حيث إنه يستخدم الصور ذات التدرج الرمادي التي تكسبه هذه السرعة.

ولقد ساهم في نجاح هذه الخوارزمية وإكسابها قوة ومعدلات عالية في الأداء هو اعتمادها على ثلاث أمور ذات أهمية كبيرة ورئيسية والتي هي:

1. *Integral image*:

أو ما يسمى بالصورة المتكاملة، حيث تعد الصورة المتكاملة واحدة من أهم أدوات لتسريع حساب الميزات، فقد سمحت هذه الخاصية بإكساب الخوارزمية السرعة الكبيرة في الكشف والدقة العالية. حيث هذا النظام لا يتعامل مباشرة مع شدة الصورة كما في الأنظمة الأخرى، وإنما يعتمد على استخدام مجموعة من الميزات التي تسمى *haar features* والتي سوف نستفيض في شرحها لاحقاً.

لنفترض بدايةً بأنه لدينا هذه الصورة بعد تحويلها إلى صورة رقمية:

1	2	3	4
5	6	7	8
1	2	3	4
5	6	7	8

وللقيام بحساب الصورة المتكاملة نعتمد المعادلة التالية في الحساب:

$$II(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

حيث $i(x, y)$: هي موضع البكسل في الصورة

ومن ثم نقوم بشكل متكرر بالحساب للمعادلة السابقة ليتم الحصول على جميع قيم البكسلات وفق ما يلي:

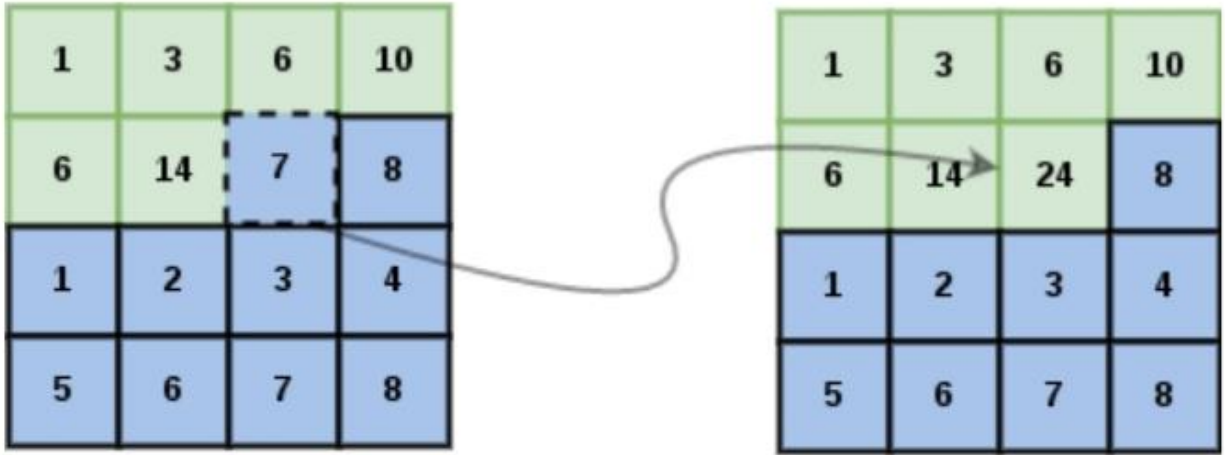
$$II(x, y) = i(x, y) - II(x - 1, y - 1) + II(x, y - 1) + II(x - 1, y)$$

(معادلة 2)

ولتوضيح الفكرة نقوم بحساب الصورة المتكاملة عند النقطة (2,1)، وذلك باستخدام المعادلة (2):

$$II(2,1) = i(2,1) + II(1,1) + II(2,0) - II(1,1) = 7 + 14 + 6 + -3 = 24$$

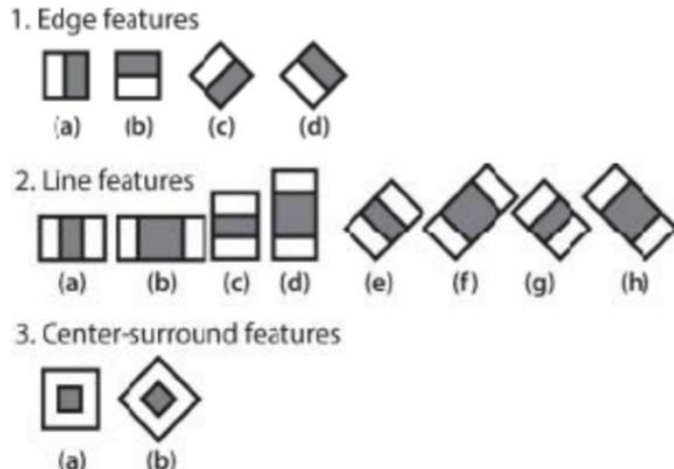
ويوضح الشكل التالي الآلية التي نحسب بها الصورة الكاملة:



حيث نعمل على تجميع البكسلات بطريقة تساعد على خفض عدد العمليات الحسابية اللازمة للحصول على مجموع شدة البكسل داخل النافذة عند حساب الميزات للصورة.

• *Haar features* :

وهي سلسلة من الوظائف مربعة الشكل ،والتي تم قياسها من قبل العالم *Alfrid Haar* في عام 1909 ، وهي تطبق بشكل مباشر في عملية كشف الوجه ،وهي تشبه *Convolution kernels* التي تدرس بشكل أساسي في *Convolution Neural Network* . وهنا سيتم تطبيق هذه الخصائص على جميع الأجزاء ذات الصلة بالوجه لتحديد الوجه البشري خاصةً.

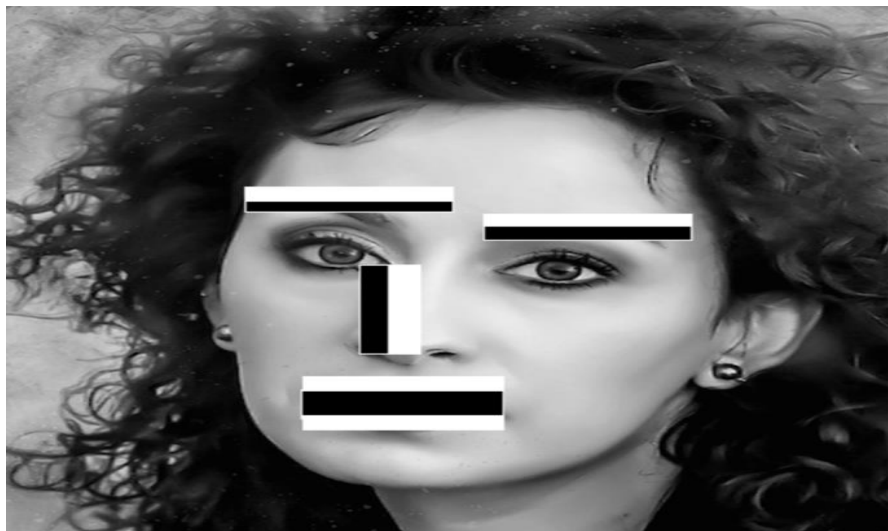


حيث كما نرى في الصورة أعلاها بأن هناك خصائص حواف وخصائص خطية وخصائص مركزية. وقد تم اعتماد هذه الخصائص للوجوه البشرية بسبب تشاركها في بعض الميزات ،فمثلاً لو نظرت إلى صورة شخص ما ستجد بأن منطقة العين أعمق من جسر الأنف ، كما أن الخدين هم أكثر إشراقاً من منطقة العين ، وما إلى هنالك من هذه الخصائص ..

تحتوي الصورة التالية على وجه بشري وسنرى كيف سيتم تطبيق تلك الخصائص المذكورة عليها بعد تحويلها إلى صورة رمادية وذلك لتسهيل التعامل معها من ناحية المعالجة وتوفير الوقت .



وسنطبق الخصائص الخطية لكشف كل من الحواجب والأنف وخصائص الحواف لكشف الفم كما في الشكل التالي:



ولكن يتم حساب هذه الخصائص بشكل رقمي ؟!... وهذا ما سنتطرق إليه .

تكون في الحالات المثالية الصورة عبارة عن بكسلات تأخذ القيمة 0 أو 1 ولكن هذا ليس موجود على

الإطلاق ،لأن الصورة قد تم تحويلها إلى تدرجات اللون الرمادي، كما توضحه الصورة التالية.

0	0	1	1
0	0	1	1
0	0	1	1
0	0	1	1

0.1	0.2	0.6	0.8
0.3	0.2	0.6	0.8
0.2	0.1	0.8	0.6
0.2	0.1	0.8	0.9

قيم البكسل: 0 و 1 للحالة المثالية (الصندوق العلوي) ، قيم تدرج الرمادي الطبيعي للحالات الحقيقية (الصندوق السفلي)

ولنفترض على سبيل المثال بإننا نريد أن نقوم بكشف الحاجب ،فإننا سنقوم بجمع البكسلات الموجودة في

المنطقة الفاتحة ونجمع أيضا البكسلات الموجودة في المنطقة الداكنة ومن ثم نطرح المجموعين من بعضهما

حسب هذه العلاقة:

$$\Delta = \text{dark} - \text{white} = \frac{1}{n} \sum_{\text{dark}}^n I(x) - \frac{1}{n} \sum_{\text{white}}^n I(x)$$

ومن ثم نقارن الناتج لنعرف فيما إذا كانت قد تحققت هذه الميزة أم لا...

ففي الحالة المثالية يكون الناتج :

$$\Delta = (1) * (1/8) - (0) * (1/8) = 1$$

أما في الحالة العملية يكون الناتج:

$$\Delta = (5.9) * (1/8) - (1.3) * (1/8) = 0.575$$

ولنفترض هنا بإننا وضعنا عتبة بحيث أي قيمة أقل من أو تساوي 0.3 تعتبر ببيضاء ،وأي شيء أكبر من 0.3 يعتبر داكناً... وهكذا نضع عتبة لكل خاصية على حدى ليتم كشفها ...

: AdaBoost

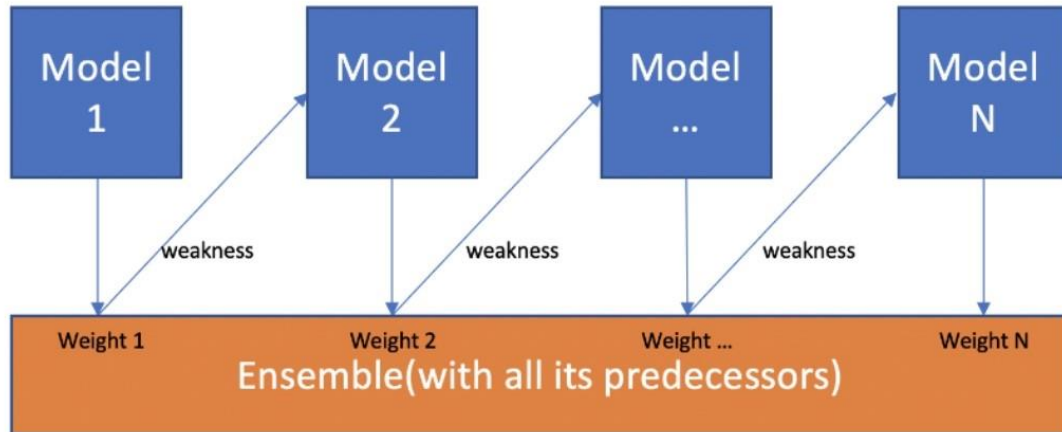
تعتبر هذه الخوارزمية ثاني أهم عامل في إنجاح تقنية فيولا-جونز ،وهي من أحد خوارزميات التعلم الآلي التي تم اعتمادها من قبل *Yoav Freund* و *Robert Schaoire* تقوم هذه الخوارزمية على بناء مصنف قوي اعتماداً على مصنفات جزئية ضعيفة ... ولكن ماذا يعني كلاً من مصنف قوي ومصنف ضعيف...؟!

المصنف القوي : هو التصنيف الذي لديه قدرة عالية على تحديد الغرض الذي تم إنشائه من أجله مع احتمالية الخطأ ضعيفة جداً.

المصنف الضعيف : هو التصنيف الذي يكون أدائه محدود جداً ،وليس لديه القدرة على التصنيف بشكل صحيح واحتمال الخطأ يكون كبير نسبياً.

حيث نقوم بإنشاء أول مصنف جزئي ونقوم بتدريبه ، ومن نقوم بإنشاء مصنف ثاني يعتمد في تدريبه على أخطاء المصنف الذي قبله ،وهذا من أجل التقليل من عدد الأخطاء الذي قد وقع بها المصنف السابق ،ونستمر في إضافة المصنفات حتى نصل إلى أقصى حد من الممكن إضافته ،ومع أقل نسبة خطأ ممكنة . فإن الهدف الأساسي من *Boosting* هو تقليل الخطأ والوصول إلى دقة عالية .

ويوضح هذا الشكل بأنه عند إنشاء النموذج الأول وتسجيل الأخطاء من النموذج الأول بواسطة خوارزمية التدريب ، يتم إعطاء سجل الأخطاء من النموذج الأول إلى مدخل النموذج الثاني ، وتكرر هذه العملية حتى يتم استيفاء جميع النماذج الموجودة كما توضحه الصورة التالية :



:*Cascade Classifier*

وبعدما يتم تدريب النماذج التي تحتوي على مجموعة من الصور الموجبة (صور تحتوي على

وجه) والصور السالبة (صور لا تحتوي على وجه) وإنشاء مصنفات قوية من

خوارزمية *AdaBoost* .

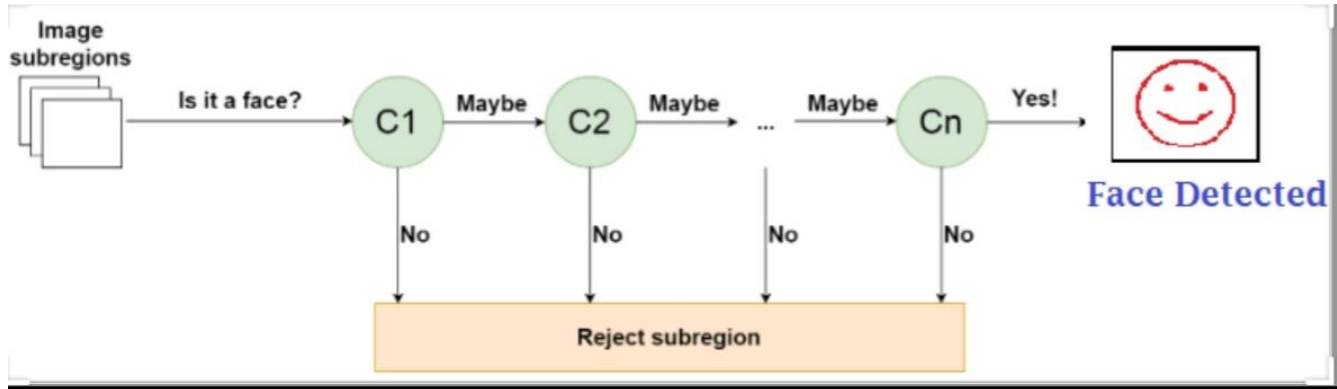
ويتم تجميع عدة مصنفات اختبار على شكل متسلسل لتمنح الخوارزمية الدقة العالية ، والتقليل من

الخطأ قدر الإمكان .

فعند دخول *blocks* من الصورة عند أول مرحلة فيتم فحصها فيما إذا كانت تحتوي على

خصائص *Haar* أم لا ..

وعندما يتم رفضها تستبعد ويأتي بآخر ،وإذا تم قبولها يتم تمريرها للمصف تدريب آخر...
وهكذا..



ويوضح الشكل التالي مخطط لسير خوارزمية فيولا جونز :

Algorithm Flow



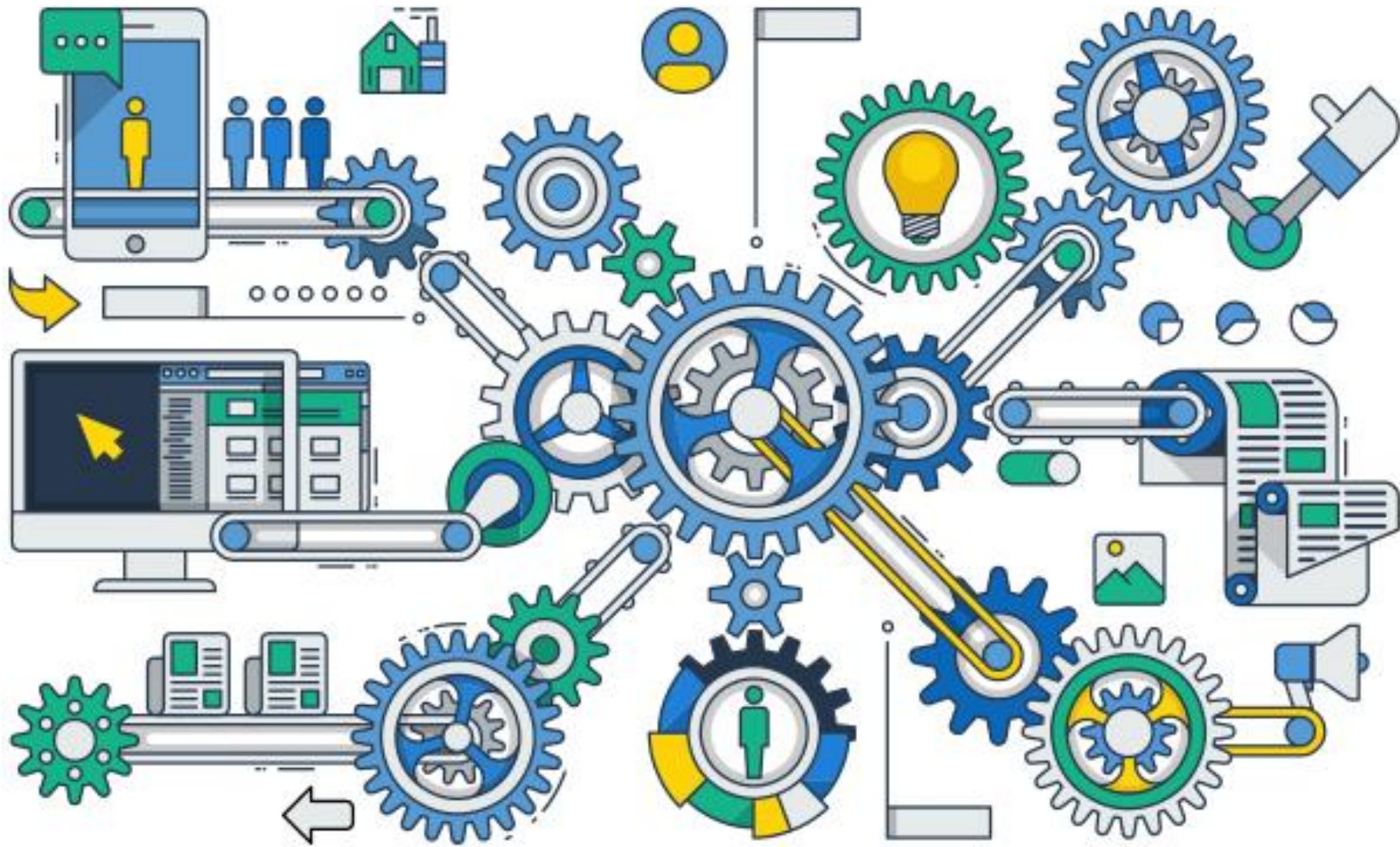
خصائص hog :

تم تطرق إلى هذه الخصائص وذلك لتحسين من جودة الكشف، حيث تعمل هذه الخاصية على مبدأ تقسيم الصورة إلى أجزاء متساوية و حساب تدرج اللون في كل جزء ويستبدل هذا المربع بأسهم لتحديد التدرج اللوني



،و هذا يفيد لتمييز الصورة التي تحتوي على وجه بكمامة او بدونها ،فالصورة الحاوية على وجه بكمامة يكون له خصائص مختلفة من حيث نوعية القماش الذي يغطي الوجه أي يكون لديه علامة فارقة بين الجلد و القماش ...

الجزء العملي



مقدمة:

تقوم فكرة المشروع على بناء نظام مشفى مؤتمت بالاستعانة بإنترنت الأشياء وباستخدام الذكاء الصناعي في الرؤية الحاسوبية، حيث يتم ربط عبر إنترنت الأشياء منصة إنترنت مركزية للمراقبة في مكتب الطبيب، ويوجد منصة مركزية للمراقبة في مكتب أمن المشفى.

ويتم رفع المعلومات التالية عبر إنترنت الأشياء:

- معلومات عن مراقبة وضع المريض:

- درجة حرارة المريض.
- قياس معدل نبضات القلب.
- قياس أكسجة دم المريض.

- مراقبة أمن الغرفة:

- استخدام RFID Reader من أجل التأكد من عدم دخول أي شخص غير

معني إلى الغرفة.

- أما عن معلومات المنصة المركزية للمراقبة في مكتب أمن المشفى، فيوجد لدينا:

- كاميرات لمراقبة الكادر الطبي والأشخاص المتواجدين في المشفى من أجل

الحفاظ على الالتزام بوضع الكمامة.

بناءً على ما سبق يتم تحقيق الأهداف التالية:

- إعلام الكادر الطبي بحالة المريض عبر منصّة إنترنت الأشياء وذلك للتقليل من احتكاك

المريض مع الكادر الطبي.

- إعلام منصّة المراقبة المركزية في حال حدوث أي خرق لقوانين المشفى من حيث ارتداء

الكمامة ودخول أي شخص غير معني إلى غرف المرضى.

ونهدف من خلال هذا النظام الإلكتروني والرقمي إلى أتمتة خدمات المشفى بغية الحفاظ على

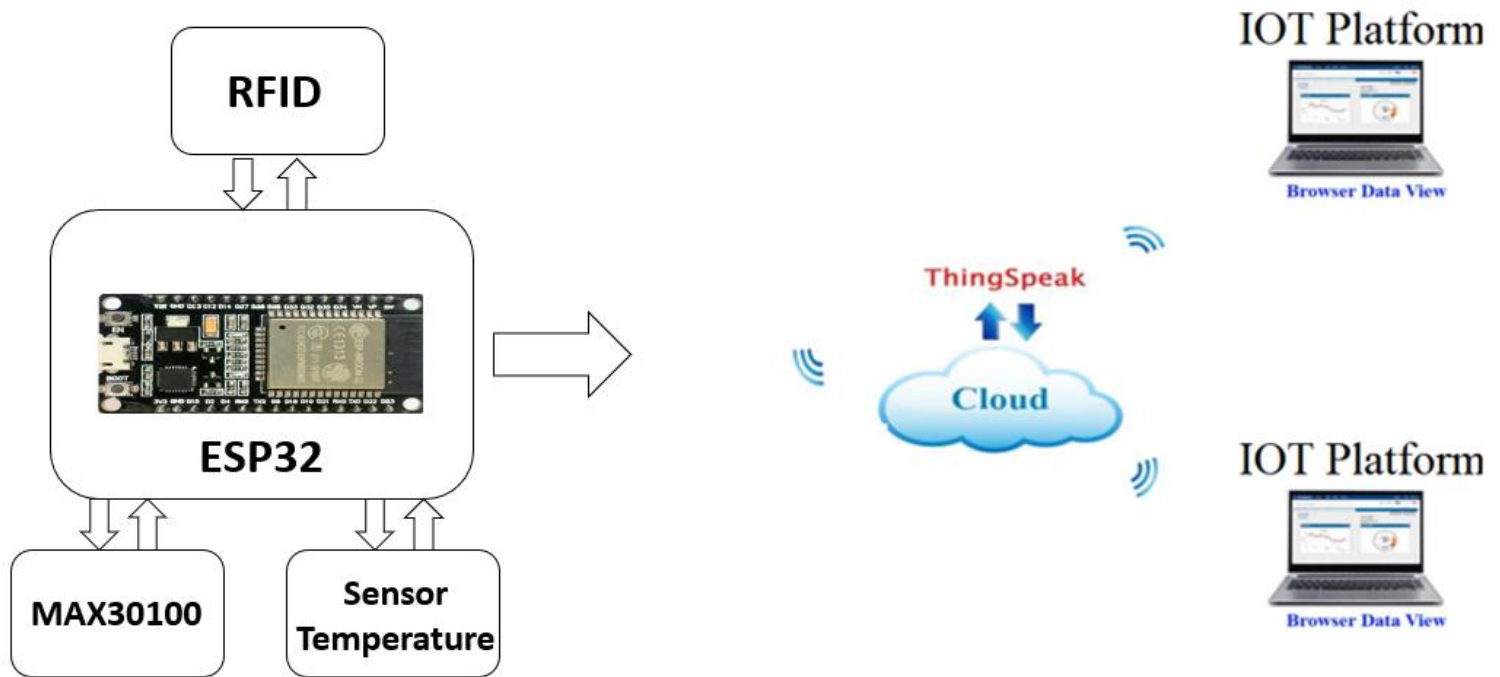
صحة كلا الكادر الطبي والمريض وتسهيل سير عمل الكادر الطبي.

المخطط الصندوقي للمشروع:

يمكن القيام بالوظائف المختلفة والربط مع الانترنت باستخدام الشريحة ESP32.

يمكننا بناء المخطط الصندوقي للمكونات الالكترونية العتادية والبرمجية التي يجب استخدامها

كما يلي:



معلومات الكادر الطبي:

يتم تزويد كل ممرض و طبيب ببطاقة RFID لها رقم خاص بها ID وهي قابلة للقراءة والكتابة بشكل موثوق وآمن جداً، نقوم ببرمجتها من أجل السماح للدخول إلى غرفة المريض.

يتم ذلك باستخدام بطاقة ESP32 لعمليات التحصيل والتخاطب مع قارئ RFID ذو النوع

.MFRC522

الرّبط مع الانترنت:

نريد رفع معلومات وضع المريض إلى منصة IOT حيث يمكننا استخدام شريحة الربط مع الانترنت ESP32 المزودة بشريحة wi-fi للاتصال بشبكة محلية والدخول إلى موقع منصة IOT. يتم ذلك من خلال برمجة الشريحة وإدخال الشبكة المحلية وموقع المنصة وكلمات المرور الخاصة.

المنصات الممكن استخدامها:

- Node Red
- Thing speak
- Cloud ino
- Atmosphere
- Blynk

يمكننا استخدام منصة Node Red للربط مع نظام الحافلة وعرض المعلومات. نقوم بإنشاء قناة خاصة ضمن المنصة ذات مفاتيح كتابة وقراءة خاصة.

الرؤية الحاسوبية:

يعتمد التصوير الرقمي بشكل أساسي على معالجة الصورة و تحليلها و استخراج المعلومات المفيدة منها، وتهدف إلى بناء تطبيقات ذكية قادرة على فهم محتوى الصور كما يفهمها الإنسان، وتعتبر مرادف إلى الرؤية الفيزيولوجية، وبكرنها تحاكي ذكاء الإنسان فلا بد من دخول الذكاء الاصطناعي لتطويرها وتحقيق غاياتها ،ولهذا تم إدراج التعلم الآلي وخوارزمياته في تطبيقات الرؤية الحاسوبية وفي تطويرها وهو يعتبر من أحد أفرع الذكاء الاصطناعي .

وقد تم استخدام خوارزمية فيولا جونز لتحقيق غاية كشف الوجه في حال وجود كامرة أم لا، والتي تعتمد في طياتها على بنى أساسية في التعلم الآلي و التي هي تحصيل البيانات ومن ثم تدريبها و يليها اختبار النتائج ، وهذا يعني أنه يتوجب علينا فرز البيانات قبل معالجتها و

قسمها إلى جزئين ويكون الجزء الأول هو مجموعة من الصور التي تحتوي على وجوه بدون كامرة و أما الجزء الآخر فيحتوي على صور لوجوه مع كامرة ومن ثم يقوم على تدريبها وتحديد فاصل العتبة، ومن بعدها يقوم باختبار النتائج التي توصلها وتحدد الدقة على إثرها. وتعتمد أيضا هذه الخوارزمية على خصائص Haar التي تعتبر من أهم الخطوات لكشف الوجه بشكل عام، وأما لكشف الكامرة فتم استخدام خصائص Hog.

جرى العمل وفق المراحل التالية :

- دراسة بحثية عن IOT والتقنيات المرتبطة.
- تأمين المكونات الالكترونية واختبارها جزئياً.
- تصميم منصة IOT بعد اختيار أحد منصّات Cloud.
- دراسة بحثية عن Computer Vision والتقنيات المرتبطة بها.
- برمجة الكاميرا الحاسوبية.
- مكاملة الأجزاء والاختبارات النهائية.

البيئة البرمجية Pycharm باستخدام لغة Python:

- هو بيئة تطويرية متكاملة تستخدم في برمجة الحاسوب، وخاصة البرمجة بلغة البايثون، وهو من نتاج الشركة التشيكية جيت برينز JetBrains وهي شركة معروفة في تطوير الكثير من بيئات التطوير، ومما يدل على قوة هذه البيئة لمستخدميها.
- ويعد بيئة عمل مثالية تمكن المبرمج من إتمام مهامه التي يقوم بها في البايثون، حيث يمتلك مواصفات عالية ورائعة تغطي حاجة المبرمج وكما تأتي أيضا بكل الأدوات اللازمة، إذ أنه يقوم باستدعاء جميع الملفات والمكتبات الخاصة باللغة و يسهل على المبرمج البحث عنها الاقتراحات التي يريد كتابتها أو يبحث عنها، مع إعطاء أيقونة البحث عن الملفات و الأدوات و كل ما يلزم، وكما يدعم جميع أنواع قواعد البيانات ويتيح البرنامج تحليل الكود وكشف الأخطاء.

شرح أجزاء المشروع:

شريحة التحكم ESP32:

بغية توفير الوقت والجهد، فضّلنا الاعتماد على دارة جاهزة متوفرة في الأسواق تقوم بمهمة وحدة المعالجة ولتسهيل عملية وصل المغارز إلى المنافذ المطلوبة وهذه الدارة هي ESP32.

إن لوحات ESP32 هذه أعادت الصناعة المؤتمتة إلى الحياة بسبب سهولة استخدامها حيث يمكن لأي شخص لديه خلفية تقنية قليلة أو معدومة أن يبدأ بتعلم بعض المهارات الأساسية لبرمجة وإدارة اللوحة.

لهذه المتحكمات عدة أنواع: ESP32-WROOM-32 و ESP32-WROOM-32D و

ESP32-WROVER و ESP32-SOLO-1.

تعمل كل هذه الألواح بشكل مشابه بطريقة أو بأخرى، هنالك بعض الميزات الأساسية التي تجعلها مختلفة عن بعضها البعض مثل تصميم PCB، الحجم، وعدد المغارز التماثلية.

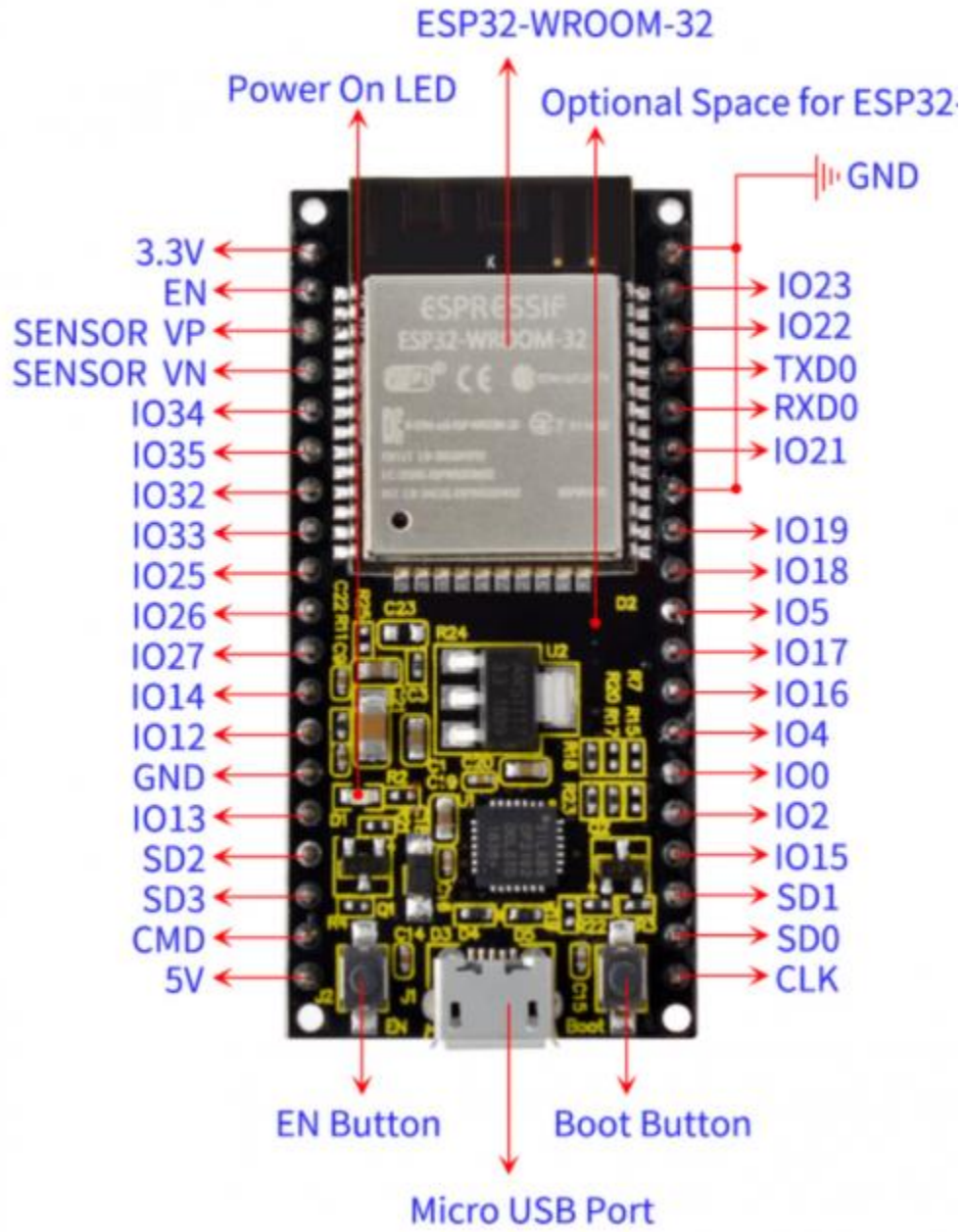
أما من الناحية البرمجية لهذه الألواح، تتم برمجة جميع هذه اللوحات في برنامج Arduino IDE، كما أنّ هذه الألواح لا تحتاج لإرفاق مكونات أو أجهزة إضافية لوضعها في حالة التشغيل.

قمنا في مشروعنا باختيار لوحة ESP32 وذلك لحجم ذواكرها وعدد مغارزها Pins I/O المناسب،

حيث يعطي تصميم مشروع باستخدام ESP32 مرونة في العمل مع مساحة ذاكرة أكبر وقوة

معالجة تسمح لك بالعمل مع عدد من أجهزة الاستشعار في وقت واحد.

يبين الشكل (2-2) أهم مكونات شريحة ESP32:



القارئ المستخدم RC522 RFID:

قمنا باختيار القارئ RC522 ذو الدقة الجيدة والسعر المناسب، يعمل هذا القارئ الموضح في

الشكل على جهد 5 فولط، ويدعم ثلاث بروتوكولات تراسل هي:

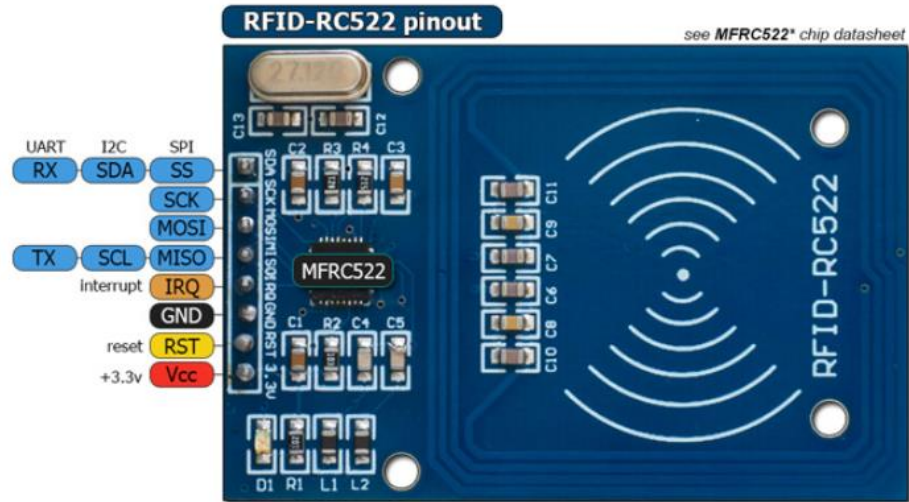
- البروتوكول UART عبر المدخلين RX و TX.

- البروتوكول المتزامن SPI عبر المداخل SS, SCK, MOSI, MISO.

- البروتوكول المتزامن I2C عبر SDA و SCK.

نختار البروتوكول SPI المتزامن وفيه خطّين للمعطيات وخط مؤقت clock، وهو ذو سرعة كبيرة

ومتوافق مع بيئة Arduino.



التوصيلات الالكترونية للشريحة ESP32:

RC522 RFID Reader	EPS32 (VSPI interface)
SDA	5
SCK	18
MOSI	23
MISO	19
IRQ	not connected
GND	GND
RST	27
3.3V	3V3

حساس نبضات القلب والأكسجة MAX30100:

موديول حساس قياس نبضات القلب وقياس الأكسجة SpO2 ذو خرج رقمي عبر واجهة I2C.

مواصفات MAX30100:

- جهد التغذية 1.8 أو 3.3 فولط.

- معتمد على الدارة المتكاملة MAX30100.

- حساس دقيق لتغيرات النبض ومضبوط.

- الخرج رقمي عبر واجهة I2C.

- موديول 7 أرجل تتضمن واجهة I2C و INT و IRD و RD.

- يتضمن معالج إشارة مضمن لتقليل الضجيج ومعالجة الإشارة بشكل جيد.

مبدأ العمل:

يحتوي الجهاز على مصباحي LED ، أحدهما يصدر ضوءًا أحمر ، والآخر ينبعث منه ضوء

الأشعة تحت الحمراء . لحساب معدل نبضات القلب ، هناك حاجة فقط إلى ضوء الأشعة تحت

الحمراء .

يستخدم كل من الضوء الأحمر والأشعة تحت الحمراء لقياس مستويات الأكسجين في الدم .

عندما يضخ القلب الدم ، تحدث زيادة في الدم المؤكسج نتيجة زيادة الدم . مع استرخاء القلب ،

ينخفض أيضًا حجم الدم المؤكسج. من خلال معرفة الوقت بين زيادة ونقص الدم المؤكسج ،
يتم تحديد معدل النبض. اتضح أن الدم المؤكسج يمتص المزيد من ضوء الأشعة تحت الحمراء
ويمرر المزيد من الضوء الأحمر بينما يمتص الدم غير المؤكسج الضوء الأحمر ويمرر المزيد
من ضوء الأشعة تحت الحمراء. هذه هي الوظيفة الرئيسية لـ MAX30100 فهي تقرأ
مستويات الامتصاص لكل من مصادر الضوء وتخزينها في مخزن مؤقت يمكن قراءته عبر
I2C.



مميزات مقياس التأكسج النبضي MAX30100 :

1- يستهلك طاقة منخفضة جدًا (تعمل من 1.8 فولت و 3.3 فولت)

2- تيار الإغلاق المنخفض للغاية ($0.7\mu A$)

3- سرعة إخراج البيانات

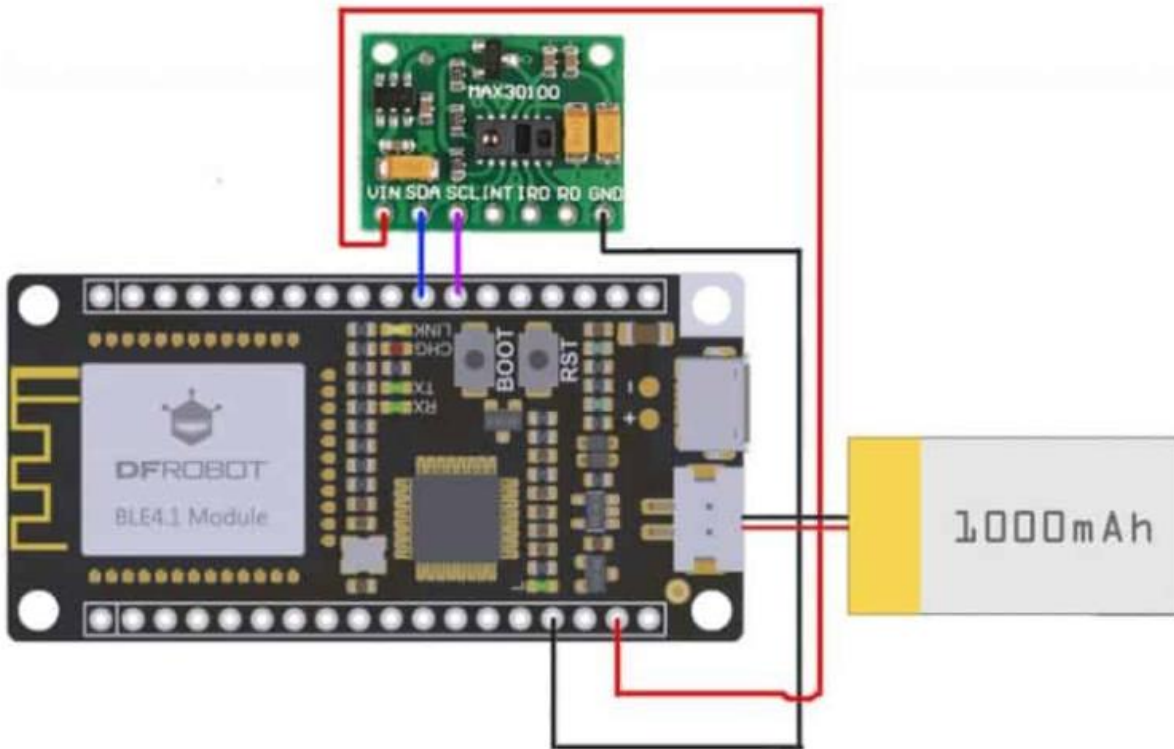
كيفية التوصيل:

يحتوي MAX30100 على دبابيس I2C ، لذا نقوم بتوصيل دبوس SDA الخاص به بمسمار

D21 و SCL بـ D22 من لوحة ESP32 ، مصدر الطاقة المطلوب بواسطة

MAX30100 هو 3.3 فولت، لذا نقوم بتوصيل VCC الخاصة بها بـ 3.3 فولت من

ESP32.



حساس الحرارة NTC 103:

الحساس الحراري ntc sensor هو عبارة عن مقاومة حرارية تتغير قيمتها الاومية بالتغير في درجة الحرارة وتعرف بأسم الثيرمستور ويوجد منها نوعان وهما المقاومة الحرارية الموجبه ptc والمقاومة الحرارية السالبة ntc

تأثير درجة الحرارة على المقاومة

تؤثر درجة الحرارة على قيمة المقاومة. فالمعادن النقية كالنحاس والألمونيوم تزيد مقاومتها بإرتفاع درجة الحرارة وذلك لأن درجة الحرارة تسبب زيادة طاقة الالكترونات الحرة فتزيد سرعتها ولذلك يزداد تصادمها مع أيونات المعدن فتزداد مقاومتها لمرور هذه الالكترونات .

أما الكربون والمحاليل وأشباه الموصلات فتقل مقاومتها بإرتفاع درجة الحرارة وهناك بعض السبائك مثل المنجانيين (85% نحاس + 12% منجنيز + 3% نيكل)

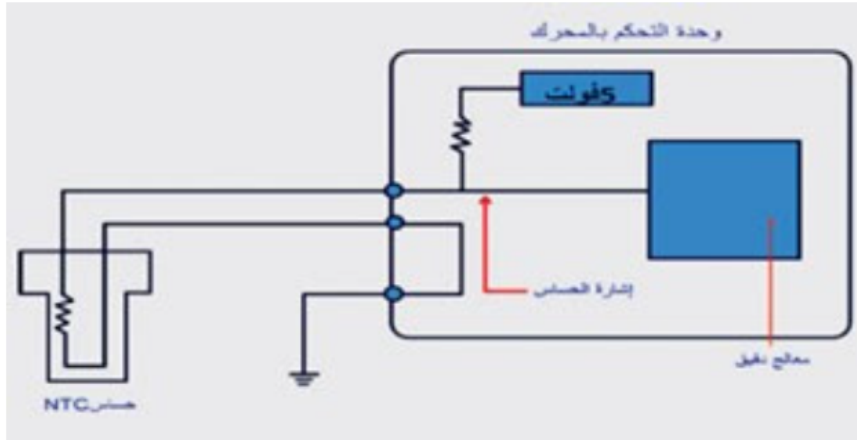
والكونستنتان(0٨% نحاس أحمر , ٤١% نيكل , ١% منجنيز) فتتغير مقاومتها تغييرا صغيرا نسبيا ويمكن إهمال هذا التغير عند حد معين لدرجة الحرارة ولذلك تستخدم مثل هذه السبائك فى صنع المقاومات القياسية.

يمكن التعبير عن تغير المقاومة بتغير درجة الحرارة بمعامل يطلق عليه إسم المعامل الحراري للمقاومة. بمعنى هناك مقاومات تزيد مقاومتها بإرتفاع درجة الحرارة واخري تقل لنفس السبب وتم استخدام هذه المقاومات كحساسات .

المقاومة ذات المعامل الحراري السالب NTC

ويرمز لها بالرمز (NTC) وهي الأكثر شيوعا، وتمتاز بنقصان مقاومتها عند ارتفاع درجة حرارتها وتزيد مقاومتها بانخفاض درجة حرارتها، وتستخدم هذه المقاومات في العديد من الحساسات، ولهذه المقاومات استطاعات مختلفة تناسب الأداء المطلوب.

من خلال التغير لمقاومة هذا الحساس بتغير درجة حرارته فإنه يستفاد من هذه الظاهرة في قياس درجات الحرارة لمكونات مختلفة في المركبة (حرارة سائل التبريد، حرارة الهواء، حرارة الوقود....)، ولهذا الحساس نهايتان ترتبطان مع وحدات التحكم، النهاية الأولى تتصل بخط سالب والنهاية الثانية تتصل بخط موجب، ويدعى أيضا خط إشارة



وتستخدم هذه الأنواع من المقاومات في أجهزة القياس الكهربائية ومحولات الطاقة الصغيرة المستخدمة لقياس الحرارة بشكل عام،



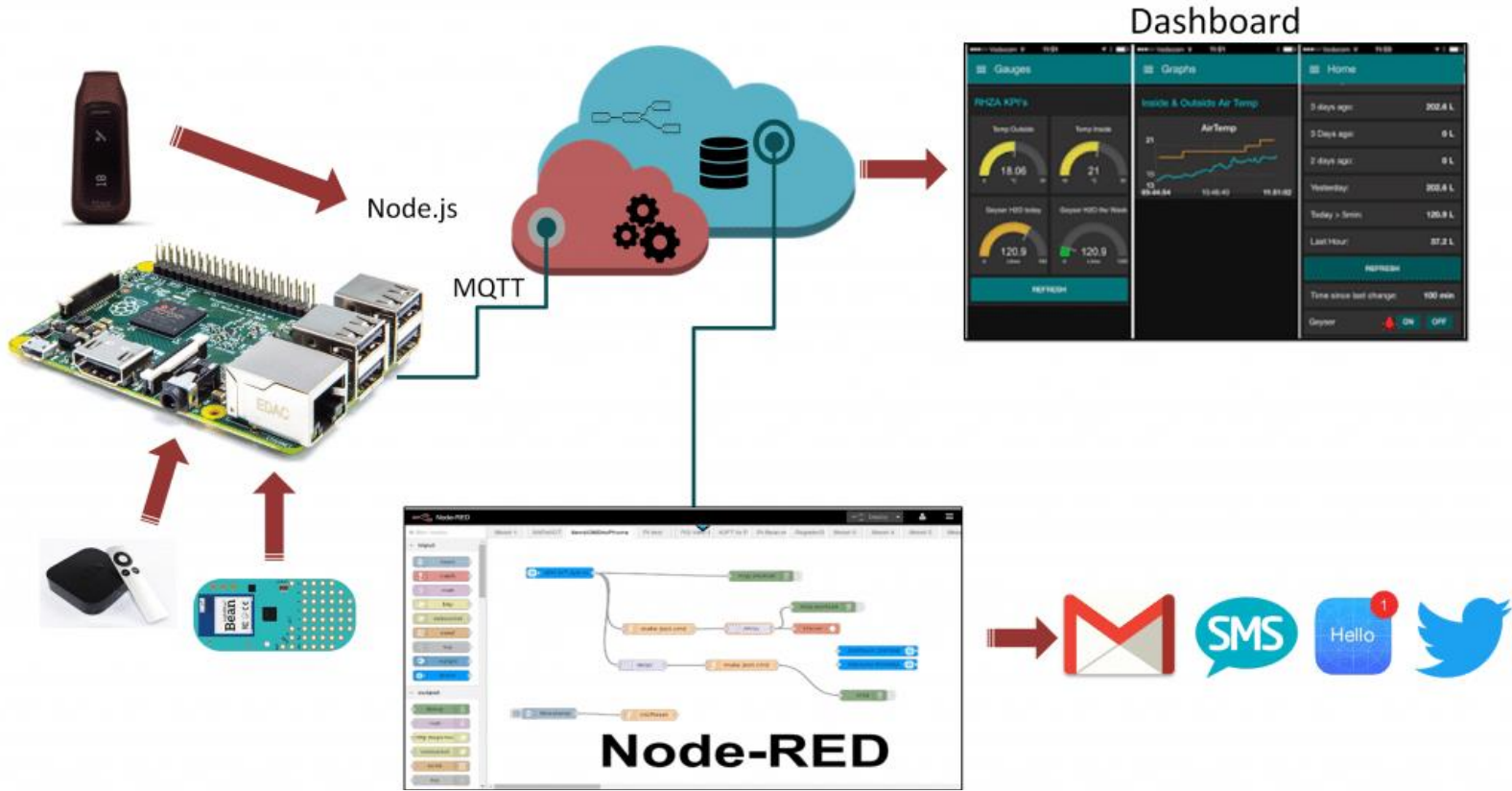
أما في السيارات فتستخدم في عدة اماكن منها علي سبيل المثال تستخدم كعنصر حساس للحرارة لقياس درجة حرارة ماء التبريد بالنسبة لمحركات السيارات (ماء الرادياتير) عن طريق ما يسمى بأمبير الحرارة.

حيث عندما تسخن المياه تسخن أيضا مقاومة NTC فتقل قيمة المقاومة الحرارية ويسري التيار في جهاز القياس وكلما زادت الحرارة كلما قلت المقاومة وازدادت قراءة جهاز القياس . أي أننا في هذه الحالة قمنا بتحويل الاشارات الحرارية إلى إشارات كهربية يمكن التعرف عليها من خلال جهاز القياس.

منصة IOT المنفذة:

Node-RED هي أداة قوية لبناء تطبيقات انترنت الأشياء مع التركيز على تبسيط توصيل الأسلاك معا بين blocks التعليمات البرمجية لتنفيذ المهام. حيث يستخدم نهج البرمجية المرئية التي تسمح للمطورين لربط blocks التعليمات البرمجية المحددة مسبقا، و المعروفة باسم العقد "nodes"، معا لأداء المهمة. العقد "Nodes" المتصلة ، عادة ما تكون مزيجاً من عقد الادخال (input nodes)، عقد المعالجة (processing nodes) ، و عقد الاخراج (output nodes)، يتم توصيلهم معا لتشكيل التدفقات. "Flows"

في الأصل تم تطويرها كمشروع مفتوح المصدر من قبل IBM ، لتلبية حاجتهم للاتصال السريع بين الأجهزة و خدمات الويب ((web services و البرامج الأخرى، وسرعان ما تطورت لتكون الغرض العام لأداة برمجة . IOT وعلى الرغم من أن Node-RED مصممة أصلا للعمل مع انترنت الأشياء ، أي الأجهزة التي تتفاعل و تتحكم بمحيطها، كما تطورت و أصبحت مفيدة لمجموعة واسعة من التطبيقات.

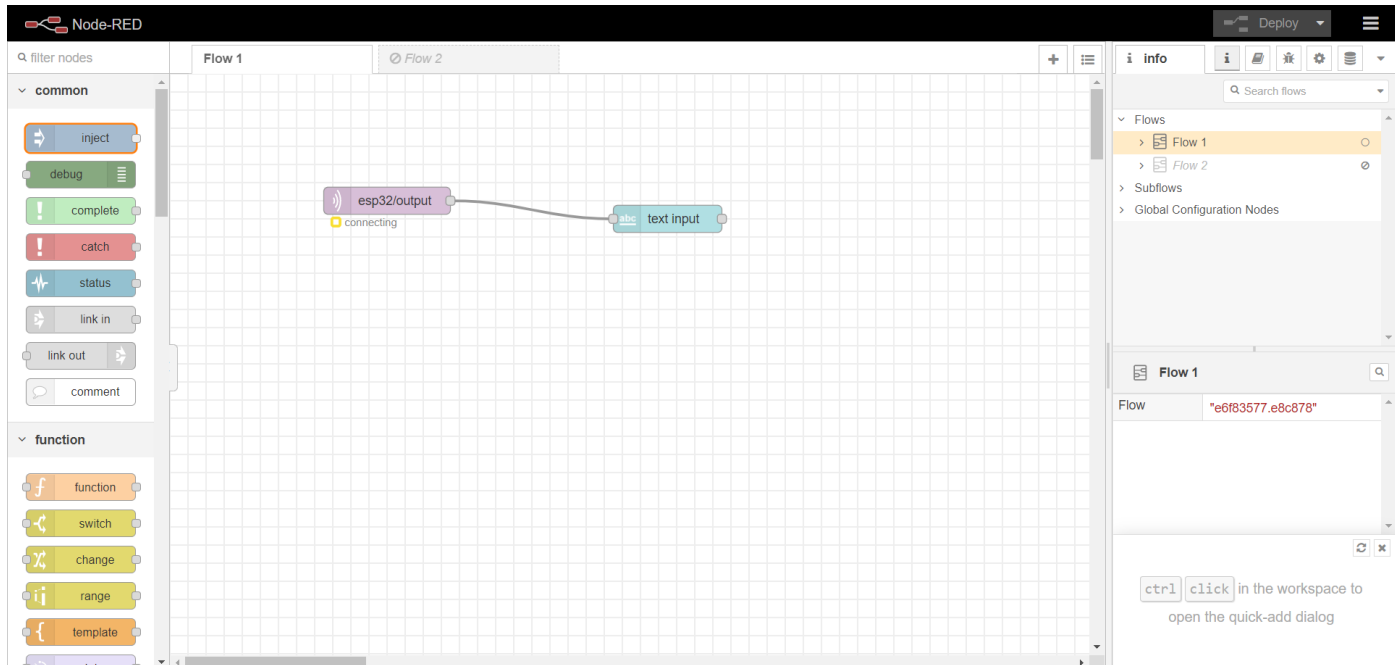


الـ Node-RED هي مثال لنموذج البرمجة القائمة على التدفق - الرسائل تمثل الأحداث التي تتدفق بين العقد ثم يتم معالجتها لتؤدي إلى نتائج. نموذج البرمجة القائم على التدفق هو رسم تفصيلي واضح وجيد لتطبيقات انترنت الأشياء النموذجية Node-RED. تقوم بحزم أحداث العالم الحقيقي كرسائل تقدم نموذجاً بسيطاً و موحد للأحداث و التي تتدفق بين العقد التي تشكل التدفقات.

مجموعة العقد المدمجة هي الجزء الثاني من قصة الـ Node-RED فمن خلال بناء مجموعة من عقد الإدخال و الإخراج القوية . و التي تخفي الكثير من تعقيدات التفاعل مع العالم الحقيقي ، Node-RED تتيح للمطورين بناء لبنات قوية لتمكينهم من وضعهم معاً في تدفق لتحقيق الكثير ، دون الحاجة إلى قلق حول تفاصيل البرمجة.

هذان العاملان جعلتا من Node-RED أداة قوية لتطوير تطبيقات انترنت الأشياء.

آلية العمل والمخطط التدفقي لكود شريحة الربط مع الانترنت:

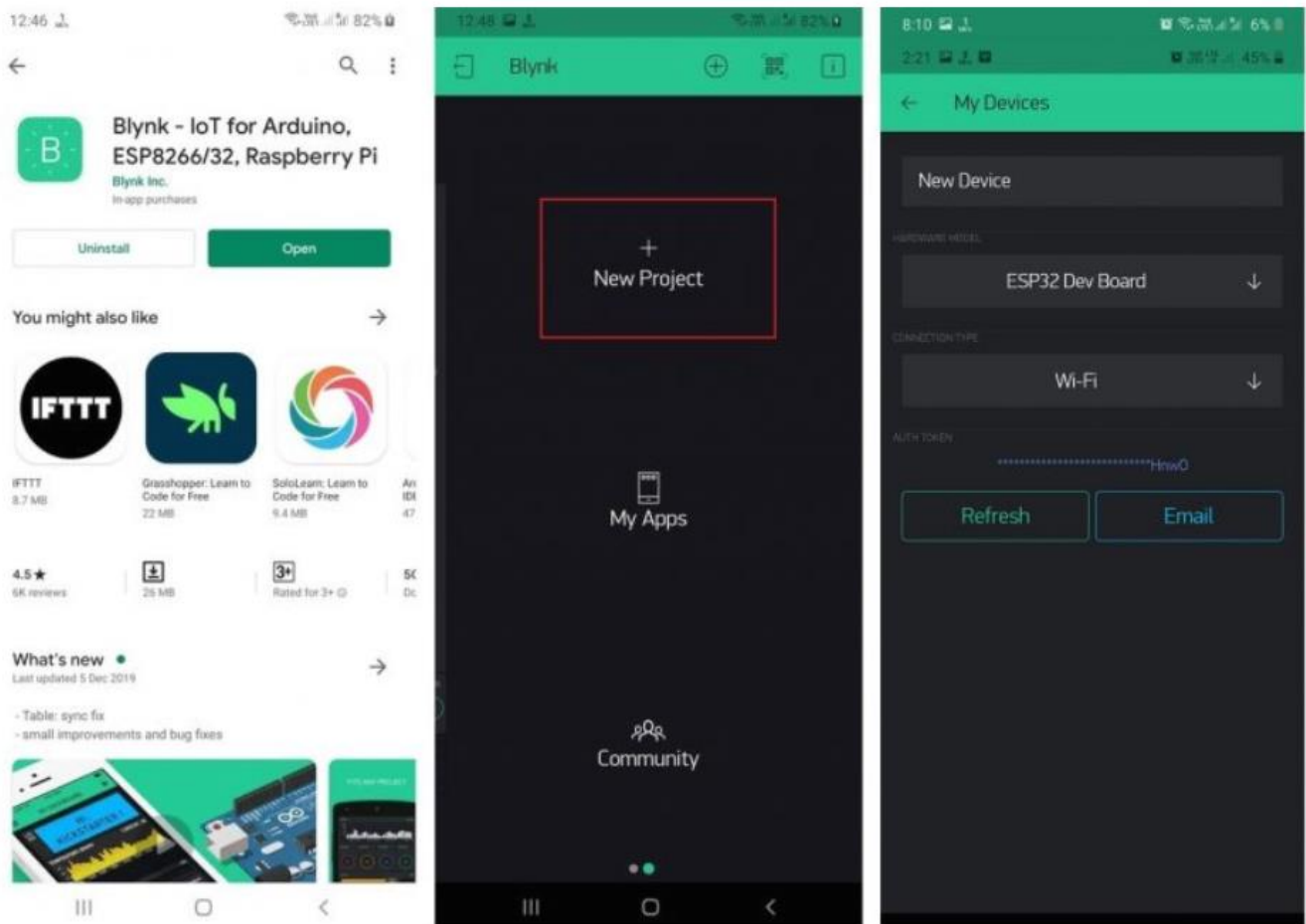


تطبيق أندرويد المنفذ :

إعداد تطبيق Blynk Android :

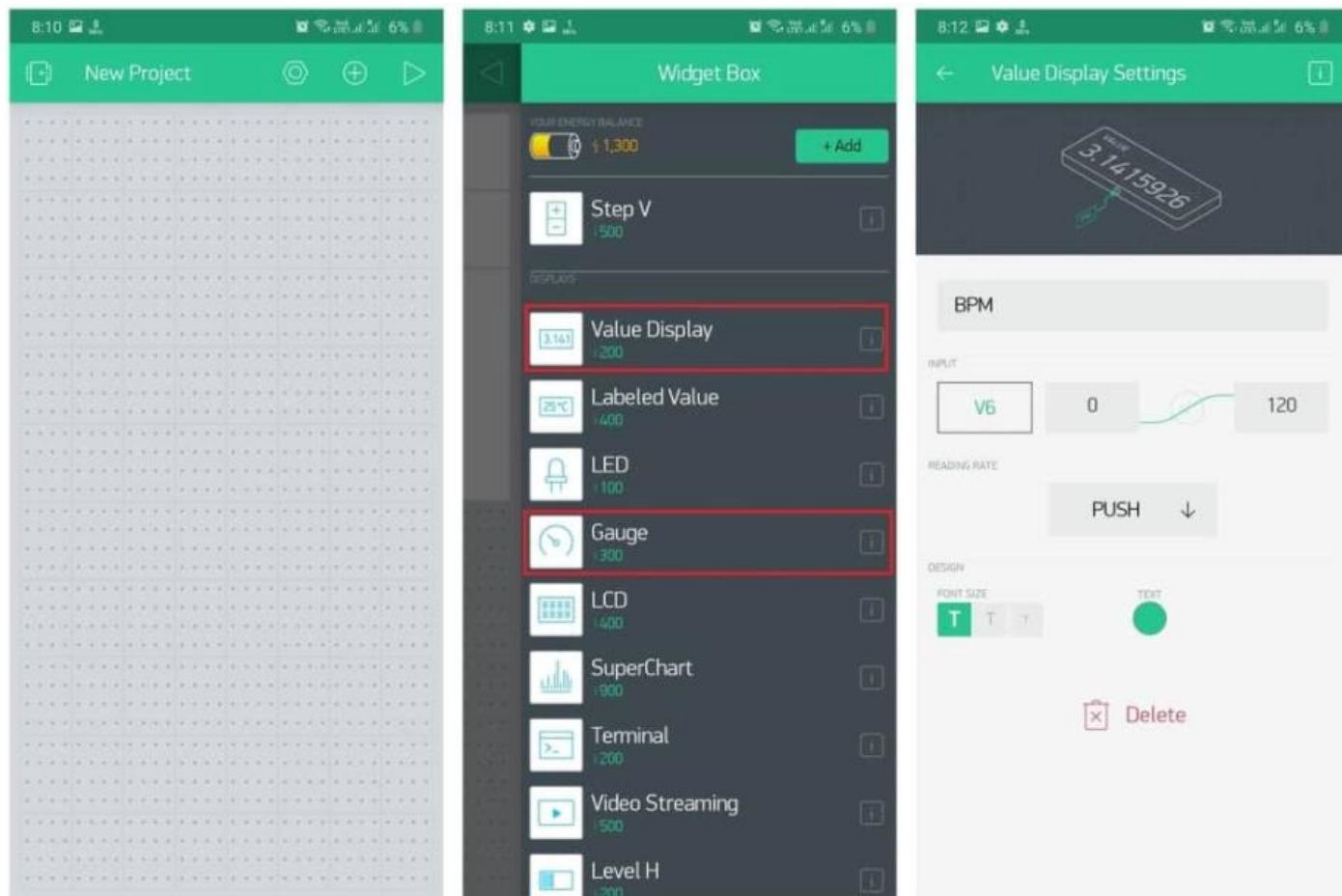
Blynk هو تطبيق يعمل على أجهزة Android و IOS للتحكم في أي تطبيق قائم على إنترنت الأشياء باستخدام الهواتف الذكية.

يتيح لك إنشاء واجهة المستخدم الرسومية لتطبيق إنترنت الأشياء. سنقوم هنا بإعداد تطبيق Blynk لمراقبة BPM & SPO2 عبر شبكة Wi-Fi باستخدام NodeMCU ESP32.

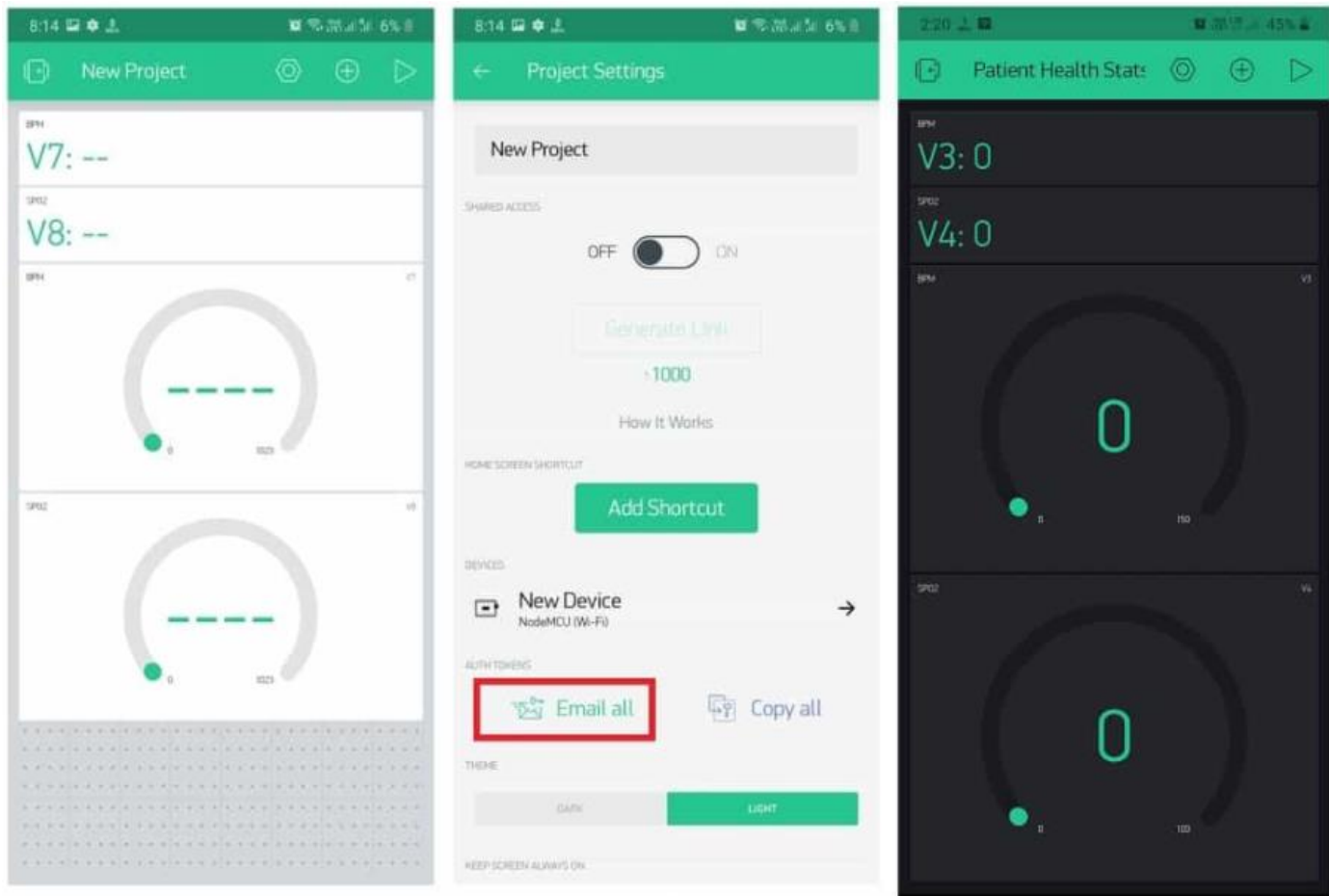


ننقر الآن على "مشروع جديد". "في النافذة المنبثقة ، نقوم بتعيين المعلومات مثل اسم المشروع واللوحة ونوع الاتصال كما هو موضح في الصورة أعلاه. بالنسبة لمشروع

MAX30100 ESP32 هذا ، نحدد الجهاز على أنه ESP32 ونوع الاتصال على أنه Wi-Fi. ثم ننقر فوق إنشاء.



الآن ننقر فوق علامة "+" لإضافة الحاجيات. نحن بحاجة إلى قراءة قيمة BPM & SpO2. لذا نحدد زوجًا من عناصر واجهة المستخدم المسمى Value Display & Gauge.



بعد سحب الأدوات ، نضبط معلوماتها كما هو موضح في الصورة أعلاه. ثم ننقر فوق عرض القيمة ونقوم بتعيين الدبوس على "V3" و "V4" وبالمثل ، في إعدادات جهاز القياس ، نضبط دبوس الإخراج على "V3" و "V4".

المكتبات المستخدمة:

مكتبة scikit-learn (SK learn):

مكتبة scikit-learn ينبع اسمها من فكرة أنها “SciKit” (مجموعة أدوات SciPy) ، وهي من مكتبات تعليم الآلة بلغة بايثون، وتحتوي على العديد من الخوارزميات والطرق المستخدمة في مجال تعليم الآلة مثل التصنيف Classification العنقدة

Clustering والانحدار Regression.

تخدم هذه المكتبة جانب تحليل البيانات وتعليم الآلة بشكل قوي، ولا تقتصر على ذلك، حيث تقدم إمكانيات هائلة في مجال معالجة الإشارات، ومعالجة الصور، والعمليات الحسابية المعقدة، وبالإضافة لاستخدامها في مرحلة تجهيز البيانات وتقييم النماذج.

وتم بناؤها بالاعتماد على مكتبات SciPy ، NumPy ، Matplotlib والعديد من المكتبات الأخرى .

انطلقت مكتبة scikit-learn في عام 2007 خلال برنامج Google Summer of Code

على يد مجموعة من المتطوعين على رأسهم ديفيد كورانبو وتأتي في الترتيب الثاني في موقع GitHub الشهير في قائمة مكتبات تعليم الآلة، وتُعتبر مدخلاً سهلاً لكل من يرغب في تعلم وتطبيق تعليم الآلة وعلم البيانات. تُركز مكتبة scikit-learn على نمذجة البيانات ولا تركز على طريق تحميل البيانات والتعامل معها وتلخيصها والذي هو دور مكتبي Pandas و NumPy بشكل رئيسي.

تحتوي مجموعة أدوات scikit-Learn على ذخيرة من خوارزميات التعلم الخاضعة للإشراف ، والتي تشمل النماذج الخطية المعممة: الانحدار الخطي ، وأشجار القرار ، وآلات المتجهات الداعمة ، وطرق بايز .

كما توفر هذه المكتبة بعض الخصائص التي يمكن استخدامها في مختلف التطبيقات ومنه:

- استخراج الميزة: باستخدام scikit-Learn ، يمكنك استخراج الميزات من النص والصور.
- التحقق المتقاطع: يمكن التحقق من دقة وصلاحية النماذج الخاضعة للإشراف على البيانات غير المرئية بمساعدة scikit-Learn.
- تقليل الأبعاد: باستخدام هذه الميزة ، يمكن تقليل عدد السمات في البيانات من أجل التصور اللاحق والتلخيص واختيار الميزة.
- التجميع: تسمح هذه الميزة بتجميع البيانات غير المسماة. طرق التجميع: يمكن دمج تنبؤات العديد من النماذج الخاضعة للإشراف باستخدام هذه الميزة.

مميزات مكتبة scikit-learn :

- يتم توزيع المكتبة بموجب ترخيص BSD ، مما يجعلها مجانية مع الحد الأدنى من القيود القانونية والتراخيص.
- سهولة ومرنة الاستخدام.
- مكتبة scikit-Learn متعددة الاستخدامات وسهلة الاستخدام وتخدم أغراض العالم الحقيقي مثل التنبؤ بسلوك المستهلك ، وإنشاء صور عصبية ، وما إلى ذلك.
- يتم دعم وتحديث Scikit-Learn من قبل العديد من المؤلفين والمساهمين ومجتمع دولي واسع عبر الإنترنت.

- يوفر موقع الويب scikit-learn توثيقًا تفصيليًا لواجهة برمجة التطبيقات للمستخدمين الذين يرغبون في دمج الخوارزميات مع الأنظمة الأساسية الخاصة بهم.

مكتبة OpenCv:

المكتبة البرمجية المفتوحة للرؤية الحاسوبية (opencv)

وهي مكتبة اقترانات برمجية تهدف بشكل أساسي إلى تطوير الرؤية الحاسوبية، وقامت بتطويرها شركة إنتل Intel، وهي مجانية تحت رخصة مصدر مفتوحة (open source computer vision) ويمكن استخدامها على جميع الأنظمة الحاسوبية وهي تركز بشكل أساسي على معالجة الصورة بالزمن الحقيقي real time

محتويات مكتبة opencv:

- **Core** : وحدة نمطية مدمجة لتحديد هياكل البيانات الأساسية المستخدمة من قبل جميع الوحدات أخرى.
- **Img_proc** : وحدة معالجة الصور التي تعنى بعمليات تحسين الصورة كعمليات الترشيح، والتحويلات الهندسية على الصور مثل تغيير حجم الصورة، أو تغيير التباين و سطوع الصورة وتعديل الهستوغرام للصورة والعديد من العمليات.
- **Video** : وحدة تحليل الفيديو التي تتضمن تقديرات الحركة للهدف، الخلفية والبيئة المحيطة، وخوارزميات تتبع الكائن أو الهدف.
- **Calib3d** : وحدة تحليل المعلومات للصورة ثنائية الأبعاد والتي هي مأخوذة من الأساس من بيئة ثلاثية الأبعاد وتحتوي على خوارزميات تحديد الخواص المهمة للبيئة ثلاثية الأبعاد.
- **Features2d**: وهي المسؤولة عن كشف الخصائص المعروفة للهدف وفصلها عن باقي البيئة المحيطة مثل الأشكال الهندسية مثلًا (مربع، دوائر، مستطيلات).

- **Objectdetect**: كشف عن وجوه وحالات من الفئات المحددة مسبقا (على سبيل المثال: وجوه، عيون، أشخاص، السيارات).
- **Hight_gui**: وهي واجهة سهلة الاستخدام لالتقاط الفيديو والصورة ،وبالإضافة إلى قدرات أخرى لواجهة المستخدم.
- **CPU Model**: تستخدم كامل القدرات الحسابية للنظام عن طريق استخدام قوة بطاقة معالجة الفيديو من أجل تطبيق خوارزميات opencv.

تطبيقات Opencv :

- أنظمة التعرف على الوجوه Facial recognition system .
- أنظمة التعرف على الإيماءات Gesture recognition :مثل الأوامر عن طريق الإيماءات مثل تحريك كرسي لشخص معاق عن طريق حركة العين.
- التفاعل بين الكمبيوتر والإنسان HCI.
- الروبوت النقل Mobile Robot .
- التعرف على الهدف Object Identification .
- ملاحقة الحركة Motion Tracking.

لغات البرمجة التي تدعم opencv:

مكتبة opencv مكتوبة أساسا بلغة C++، ولكنها تدعم معظم اللغات البرمجية بشكل

كامل مثل:

- Python
- Java
- Matlab
- C#
- Ruby

آفاق مستقبلية للمشروع

تكمن أهمية المشروع في إمكانية استخدامه عملياً في تسهيل العملية الطبية، يمكن تطوير المشروع ومكاملته من خلال:

- ✓ استخدام النظام لجميع أجزاء ومكونات المشفى.
- ✓ استخدام منصة انترنت أشياء مدفوعة لتسريع عمليات تحديث المعلومات.
- ✓ إجراء دراسة إحصائية لوضع المشفى بغية اتخاذ بعض الإجراءات.

خاتمة

لقد تمّ في هذا المشروع تنفيذ نظام مشفى باستخدام انترنت الأشياء والذكاء الصناعي. ونأمل أن يشكّل هذا العمل نواة لمشروع حقيقي لتطوير وتعميم أنظمة أتمتة خدمات مشفى وغيرها من عمليات الأتمتة الذكيّة بما فيه من تطوير لهذا الجانب الهام في حياتنا.

ملحق -الكودات البرمجية الأساسية

كود حساس الحرارة TMC 103:

```
which analog pin to connect //

define THERMISTORPIN 15#

resistance at 25 degrees C //

define THERMISTORNOMINAL 10000#

temp. for nominal resistance (almost always 25 C) //

define TEMPERATURENOMINAL 25#

how many samples to take and average, more takes longer //

'but is more 'smooth //

define NUMSAMPLES 5#

The beta coefficient of the thermistor (usually 3000-4000) //

define BCOEFFICIENT 3950#

the value of the 'other' resistor //

define SERIESRESISTOR 10000#

;int samples[NUMSAMPLES]

} void setup(void)
```

```

Serial.begin(9600);

{

} void loop(void)

;uint8_t i

;float average

take N samples in a row, with a slight delay //

} for (i=0; i< NUMSAMPLES; i++)

;samples[i] = analogRead(THERMISTORPIN)

;(10)delay

{

average all the samples out //

;average = 0

} for (i=0; i< NUMSAMPLES; i++)

;average += samples[i]

{

;average /= NUMSAMPLES

```

```

;Serial.print("Average analog reading ")

;Serial.println(average)

convert the value to resistance //

;average = 1023 / average - 1

;average = SERIESRESISTOR / average

;Serial.print("Thermistor resistance ")

;Serial.println(average)

;float steinhart

steinhart = average / THERMISTORNOMINAL; // (R/Ro)

steinhart = log(steinhart); // ln(R/Ro)

steinhart /= BCOEFFICIENT; // 1/B * ln(R/Ro)

steinhart += 1.0 / (TEMPERATURENOMINAL + 273.15); // + (1/To)

steinhart = 1.0 / steinhart; // Invert

steinhart -= 273.15; // convert absolute temp to C

;Serial.print("Temperature ")

;Serial.print(steinhart)

;Serial.println(" *C")

```

```
delay(1000);
```

```
{
```

كود الـ RFID :

```
include <SPI.h>#
```

```
include <MFRC522.h>#
```

```
define SS_PIN 5#
```

```
define RST_PIN 27#
```

```
.MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance
```

```
()void setup
```

```
}
```

```
Serial.begin(19200); // Initiate a serial communication
```

```
SPI.begin(); // Initiate SPI bus
```

```
mfrc522.PCD_Init(); // Initiate MFRC522
```

```
;Serial.println("Approximate your card to the reader...")
```

```
;()Serial.println
```



```

        {

        (void loop

        }

        Look for new cards //

if ( ! mfrc522.PICC_IsNewCardPresent())

        }

        ;return

        {

        Select one of the cards //

if ( ! mfrc522.PICC_ReadCardSerial())

        }

        ;return

        {

        Show UID on serial monitor//

        ;Serial.print("UID tag :")

        ;"" =String content

        ;byte letter

        for (byte i = 0; i < mfrc522.uid.size; i++)

        }

;Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ")

```

```

        ;Serial.print(mfrc522.uid.uidByte[i], HEX)

        ;content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "))

        ;content.concat(String(mfrc522.uid.uidByte[i], HEX))

    {

        ;()Serial.println

        ;Serial.print("Message : ")

        ;()content.toUpperCase

        if (content.substring(1) == "E9 45 CE 1C" ) //change here the UID of the card/cards that you
            want to give access

            }

        ;Serial.println("Authorized access")

        ;()Serial.println

        delay(3000);

        {

            else

        }

        ;Serial.println(" Access denied")

        ;()Serial.println

        }} delay(3000);

```

كود حساس القلب والأكسجة :Max30100

```
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"

#include <WiFi.h>

#define REPORTING_PERIOD_MS 1000

// Connections : SCL PIN - D1 , SDA PIN - D2 , INT PIN - D0
PulseOximeter pox;

float BPM, SpO2;
uint32_t tsLastReport = 0;

void onBeatDetected()
{
    Serial.println("Beat Detected!");
}

void setup()
{
    Serial.begin(115200);

    pinMode(19, OUTPUT);
```

```

        Serial.print("Initializing Pulse Oximeter..");

        if (!pox.begin())
        {
            Serial.println("FAILED");
            for(;;);
        }
        else
        {
            Serial.println("SUCCESS");
            pox.setOnBeatDetectedCallback(onBeatDetected);
        }

// The default current for the IR LED is 50mA and it could be changed by
// uncommenting the following line.
pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

    }

    void loop()
    {
        pox.update();

        BPM = pox.getHeartRate();
        SpO2 = pox.getSpO2();
        if (millis() - tsLastReport > REPORTING_PERIOD_MS)
        {
            Serial.print("Heart rate:");
            Serial.print(BPM);

```

```

Serial.print(" bpm / SpO2:");
Serial.print(SpO2);
Serial.println(" %");

```

```

tsLastReport = millis();
    }
}

```

كود تحليل الفيديو والمراقبة:

```

1- import os
2- import cv2
3- import numpy as np
4- from skimage.feature import hog
5- mask=[]
6- no_mask=[]
7- list_1=os.listdir("C:/Users/DELL/Desktop/mask3")
8- list_2=os.listdir("C:/Users/DELL/Desktop/no mask3")
9- l1=len(list_1)
10- l2=len(list_2)
11- face_cascade3=cv2.CascadeClassifier('D:\haarcascade_frontalface_alt.xml')
12- eye_cascade = cv2.CascadeClassifier('D:\haarcascade_eye.xml')
13- for n in range (0,l2) :
14-     im_name2=list_2[n]
15-     full_path2=os.path.join("C:/Users/DELL/Desktop/no mask3",im_name2)
16-     img2 = cv2.imread(full_path2)
17-     gray2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)
18-     faces_2 = face_cascade3.detectMultiScale(gray2, 1.3, 5)
19-
20-     for (x, y, w, h) in faces_2:
21-         # draw a rectangle in a face
22-         cv2.rectangle(img2, (x, y), (x + w, y + h), (255, 255, 0), 2)
23-         roi_gray_2 = gray2[y:y + h, x:x + w]
24-         roi_color_2 = img2[y:y + h, x:x + w,:]
25-         roi_color_2 = cv2.resize(roi_color_2 , (100, 100))
26-         roi_gray_2 = cv2.resize(roi_gray_2, (100, 100))
27-         h2 = hog(roi_gray_2, orientations=8, pixels_per_cell=(4,4), cel
ls_per_block=(1, 1), visualize=False)
28-         print(h2)
29-         no_mask.append(h2)

```

```

30- np.save("no_mask3.npy",no_mask)
31- for i in range (0,11) :
32-     im_name1=list_1[i]
33-     full_path1=os.path.join("C:/Users/DELL/Desktop/mask3",im_name1)
34-     img1=cv2.imread(full_path1)
35-
36-     gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
37-     faces1 = face_cascade3.detectMultiScale(gray1, 1.1, 1)
38-
39-     for (x, y, w, h) in faces1:
40-         # draw a rectangle in a face
41-         cv2.rectangle(img1, (x, y), (x + w, y + h), (255, 255, 0), 2)
42-         roi_gray1 = gray1[y:y + h, x:x + w]
43-
44-         roi_color1 = img1[y:y + h, x:x + w,:]
45-         roi_color1= cv2.resize(roi_color1, (100, 100))
46-         roi_gray1 = cv2.resize(roi_gray1, (100, 100))
47-         h1 = hog(roi_gray1, orientations=8, pixels_per_cell=(4,4), cell
s_per_block=(1, 1), visualize=False)
48-         print(h1)
49-         mask.append(h1)
50-         if i==2:
51-             cv2.imshow('img', roi_gray1)
52-             k = cv2.waitKey(30) & 0xff
53-             if k == 27:
54-                 break
55-             # Wait for Esc key to stop
56-         print(mask)
57- np.save("mask3.npy",mask)
58- print(n)
59- print(i)
60-

```

الكود الثاني:

```

import time
start=time.time()
import cv2
import numpy as np
from sklearn.svm import NuSVC
from skimage.feature import hog
from sklearn.model_selection import train_test_split
mask=np.load("mask3.npy")
noMask=np.load("no_mask3.npy")
sh1=mask.shape
sh2=noMask.shape

```

```

print(sh1)
print(sh2)
X=np.r_[mask,noMask]
labels=np.zeros(X.shape[0])

labels[8:]=1.0
print(labels)
names={0:'mask',1.0:'no mask'}
train_test_split(X,labels,test_size=0.25)

clff=NuSVC(nu=0.7, probability=False, decision_function_shape='ovo')
clff.fit(X,labels)
print(clff)
face_cascade3=cv2.CascadeClassifier('D:\haarcascade_frontalface_alt.xml')
face_cascade4=cv2.CascadeClassifier('D:\haarcascade_frontalface_alt.xml')
eye_cascade = cv2.CascadeClassifier('D:\haarcascade_eye.xml')

cap = cv2.VideoCapture(0)
cap.set(3,1200)
cap.set(4,1200)
cap.set(15,0.02)

while 1:

    ret, imgg = cap.read()

    gray = cv2.cvtColor(imgg, cv2.COLOR_BGR2GRAY)
    faces = face_cascade3.detectMultiScale(gray,1.1,2)

    for (x, y, w, h) in faces:
        # draw a rectangle in a face
        cv2.rectangle(imgg, (x, y), (x + w, y + h), (255, 255, 0), 2)
        roi_gray = gray[y:y + h, x:x + w]
        roi_color = imgg[y:y + h, x:x + w]
        roi_gray=cv2.resize(roi_gray,(100,100))
        h = hog(roi_gray, orientations=8, pixels_per_cell=(4,4), cells_per_block=
(1, 1), visualize=False)
        hh=h.reshape(1,-1)
        f=cv2.resize(roi_color,(100,100))
        f=f.reshape(1,-1)
        pred=clff.predict(hh)
        if pred==0:
            c=(0,255,0)

```

```

        else:
            c=(0,0,255)
            n=names[int(pred)]
            cv2.putText(imgg,n,(x,y),fontFace=1,fontScale=1,color=c,thickness=2)
            print(n)
            cv2.imshow('img', imgg)
            # Wait for Esc key to stop
            k = cv2.waitKey(30) & 0xff
            if k == 27:
                break
# Close the window
cap.release()
# De-allocate any associated memory usage
cv2.destroyAllWindows()

```

كود الصورة:

```

import time
start=time.time()
import cv2
import numpy as np
from sklearn.svm import NuSVC
from skimage.feature import hog
from sklearn.model_selection import train_test_split
mask=np.load("mask3.npy")
noMask=np.load("no mask3.npy")
sh1=mask.shape
sh2=noMask.shape
print(sh1)
print(sh2)

X=np.r_[mask,noMask]
labels=np.zeros(X.shape[0])

labels[8:]=1.0
print(labels)
names={0:'mask',1.0:'no mask'}
train_test_split(X,labels,test_size=0.25)

clff=NuSVC(nu=0.7, probability=False, decision_function_shape='ovo')
clff.fit(X,labels)
print(clff)
face_cascade3=cv2.CascadeClassifier('D:\haarcascade_frontalface_alt.xml')
face_cascade4=cv2.CascadeClassifier('D:\haarcascade_frontalface_alt.xml')
eye_cascade = cv2.CascadeClassifier('D:\haarcascade_eye.xml')

```



```

while 1:
    imgg = cv2.imread('nm1.jpg',1)
    gray = cv2.cvtColor(imgg, cv2.COLOR_BGR2GRAY)
    faces = face_cascade3.detectMultiScale(gray,1.1,2)

    for (x, y, w, h) in faces:
        # draw a rectangle in a face
        cv2.rectangle(imgg, (x, y), (x + w, y + h), (255, 255, 0), 2)
        roi_gray = gray[y:y + h, x:x + w]
        roi_color = imgg[y:y + h, x:x + w]
        roi_gray=cv2.resize(roi_gray,(100,100))
        h = hog(roi_gray, orientations=8, pixels_per_cell=(4,4), cells_per_block=
(1, 1), visualize=False)
        hh=h.reshape(1,-1)
        f=cv2.resize(roi_color,(100,100))
        f=f.reshape(1,-1)
        pred=clff.predict(hh)
        if pred==0:
            c=(0,255,0)
        else:
            c=(0,0,255)

        n=names[int(pred)]
        cv2.putText(imgg,n,(x,y),fontFace=1,fontScale=1,color=c,thickness=2)
        print(n)
    cv2.imshow('img', imgg)
    # Wait for Esc key to stop
    k = cv2.waitKey(30) & 0xff
    if k == 27:
        break
# Close the window
cap.release()
# De-allocate any associated memory usage
cv2.destroyAllWindows()

```

1. ULP Coprocessor programming - ESP32 - — ESP-IDF Programming Guide latest documentation (espressif.com) "Espressif Announces the Launch of ESP32 Cloud on Chip and Funding by Fosun Group". Espressif Systems. 2016-09-07. Retrieved 2017-03-31.
2. https://atadiat.com/ar/all-about-esp32-p1-introduction/
3. ^ "ESP32 Overview". Espressif Systems. Retrieved 2016-09-01.
4. ^ "ESP32 Datasheet" (PDF). Espressif Systems. 2017-03-06. Retrieved 2017-03-14.
5. ^ Espressif Systems (2017-08-21). "ESP32-PICO-D4 Datasheet"(PDF). Retrieved 2017-07-21.
6. ^ Jump up to:^a ^b Jim Lindblom (2016-01-21). "Enginursday: First Impressions of the ESP32". Sparkfun Electronics. Retrieved 2016-09-01.
7. ^ Jump up to:^a ^b Limor Fried (2015-12-22). Playing With New ESP32 Beta Module. Adafruit Industries. Retrieved 2016-09-02.
8. ^ Martin Harizanov (2015-12-18). "ESP32". Retrieved 2016-09-02.
9. ^ Brian Benchoff (2015-12-23). "The ESP32 Beta Units Arrive". Hackaday. Retrieved 2016-09-02.
10. ^ Markus Ulsass (2015-12-25). "ESP32 beta module HiRes pictures". Retrieved 2016-09-02.
11. ^ "FCC Part 15.247 Test Report for Espressif Systems (Shanghai) Pte. Ltd". Bay Area Compliance Laboratories Corp. 2016-02-17. Retrieved 2016-09-02.
12. ^ "ESP-WROOM-32 Datasheet" (PDF). Espressif Systems. 2016-08-22. Archived from the original (PDF) on 2016-09-13. Retrieved 2016-09-02.
13. ^ "FCC Part 15.247 Test Report for Espressif Systems (Shanghai) Pte. Ltd". Bay Area Compliance Laboratories Corp. 2016-11-10. Retrieved 2016-12-15.
14. ^ Jump up to:^a ^b "ESP-WROOM-32D/ESP32-WROOM-32U Datasheet" (PDF). Espressif Systems. Archived from the original (PDF) on 2017-12-03. Retrieved 2017-11-28.
15. ^ Baoshi (2016-10-11). "Ai-Thinker ESP-32S Decap Photos". Retrieved 2016-10-22.
16. ^ [1]
17. ^ [2]
18. ^ [3]
19. ^ Jump up to:^a ^b "ESP-32S-ALB/ALB-WROOM". AnalogLamb.
20. ^ "ALB32-WROVER is an ESP-WROOM-32 Compatible Module with 32 Mbit PSRAM, up to 128 Mbit Flash". CNXSoft.
21. ^ "(SKU:TEL0111)ESP32 WiFi&Bluetooth Module/ESP-WROOM-32". DFRobot.

- ^ ["硬件功能 \(Hardware Function\)". IntoRobot. Archived from the original on 2018-05-30. Retrieved 2017-10-02.](#) .22
 ^ ITEAD (2017-02-15). ["PSH-C32 Schematic" \(PDF\).](#) Retrieved 2017-02-23. .23
 ^ ITEAD. ["PSH-C32".](#) Retrieved 2017-02-23. .24
 ^ Pycom. ["Pycom OEM Products".](#) Retrieved 2017-03-14. .25
 ^ Jump up to:^a ^b ["NINA-W13 series". u-blox.](#) .26
 ^ William Hooi (2016-09-01). ["So looking forward..."](#). Retrieved 2016-09-02. .27
 ^ Aditya Tannu (2016-09-02). ["Look what I just got!"](#). Retrieved 2016-09-02. .28
 ^ ["ESP32-DevKitC Getting Started Guide".](#) Espressif Systems. 2016-09-21. Retrieved 2016-09-21. .29
 ^ ["ESP-WROVER-KIT".](#) Espressif Systems. Retrieved 2017-02-19. .30
 ^ ["ESP32 Camera Demo".](#) Ivan Grokhotkov. 2016-11-28. Retrieved 2016-12-02. .31
 ^ ESP32.net (2016-10-28). ["Ai-Thinker NodeMCU-32S Development Board Appears on AliExpress".](#) Retrieved 2016-10-28. .32
 ^ ["ESP32 Development Board – Developer Edition".](#) AnalogLamb. .33
 ^ ["Maple ESP32 – ESP32 Board with Micro SD Interface, USB to USART & Compatible with Arduino Interface".](#) AnalogLamb. .34
 ^ ESP32 UNO by ArduCam. ["Arduino Uno-like development board".](#) .35
 ^ ["BC24 / ESP32 Development Board – Big Circle 24".](#) SwitchDoc Labs. .36
 ^ ["Products \[WEMOS Electronics\]".](#) Retrieved 2018-01-25. .37
 ^ ["LOLIN32 \(Retired\) \[WEMOS Electronics\]".](#) wiki.wemos.cc. Retrieved 2018-11-13. .38
 ^ ["LOLIN32 Lite \(Retired\) \[WEMOS Electronics\]".](#) wiki.wemos.cc. Retrieved 2018-11-13. .39
 ^ ["LOLIN32 Pro \(Retired\) \[WEMOS Electronics\]".](#) wiki.wemos.cc. Retrieved 2018-11-13. .40
 ^ ["D32 \[WEMOS Electronics\]".](#) wiki.wemos.cc. Retrieved 2018-11-13. .41
 ^ ["D32 Pro \[WEMOS Electronics\]".](#) wiki.wemos.cc. Retrieved 2018-11-13. .42
 ^ Jump up to:^a ^b ["Third-Party Platforms That Support Espressif Hardware".](#) Espressif Systems. Retrieved 2017-10-20. .43
 ^ Tim Mattison (2017-04-13). ["AWS IoT on Mongoose OS, Part 1".](#) .44
 ^ ["Google Cloud IoT Partners".](#) Google. Retrieved 2017-10-20. .45
 ^ ["Alibaba's IoT Wrist Bands Based on ESP32".](#) Espressif Systems. 2017-09-30. .46
 ^ ["DingTalk's New Biometric Attendance Monitor Based on ESP32".](#) Espressif Systems. 2017-06-02. .47
 ^ @ESP32net (2017-11-07). ["FCC internal photos exhibit for the LIFX Mini Wi-Fi LED light \(FCC ID 2AA53-MINI\) show inclusion of ESP32..."](#) (Tweet) – via Twitter. .48

Damascus University

Faculty of Mechanical & Electrical Engineering

Electronics and Communication Engineering Department



Automating Hospital Services

A graduation project prepared to obtain a Bachelor of Science
in Electronics and Communication Engineering

Prepared by

Muhammad Moutee Al-Sabouni

Supervised by

Dr. Talal Al-Hammoud

Academic Year

2020-2021