## Development Environment:

When we implement our project we choose programming languages, IDE, SDK, Frameworks, UI tools. Details of the tools.

### Language Programming:

C++:  is a [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). It has [imperative](https://en.wikipedia.org/wiki/Imperative_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) and generic programming features, while also providing facilities for [low-level](https://en.wikipedia.org/wiki/Low-level_programming_language) [memory](https://en.wikipedia.org/wiki/Memory_(computing)) manipulation.

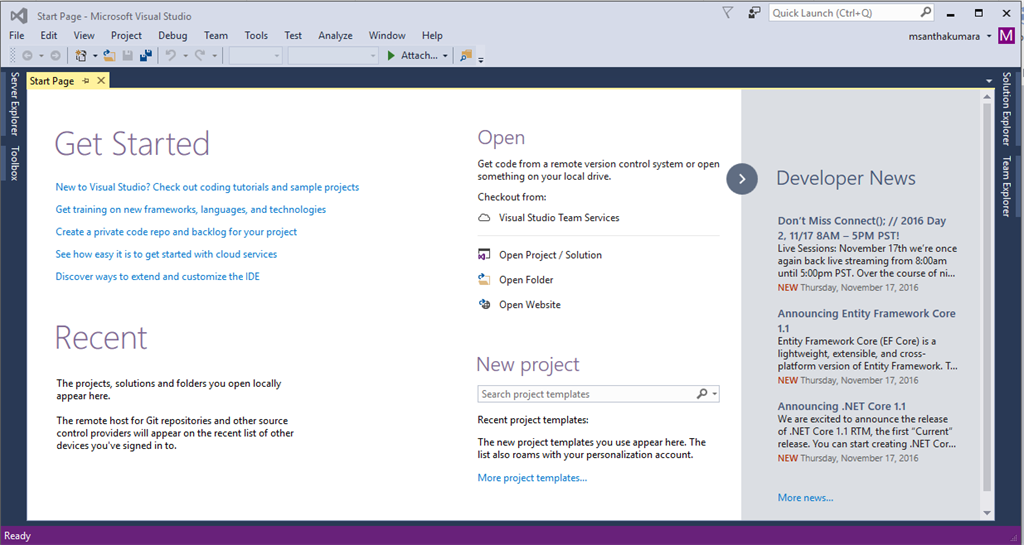
JavaScript:  is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [programming language](https://en.wikipedia.org/wiki/Programming_language). It is a language which is also characterized as [dynamic](https://en.wikipedia.org/wiki/Dynamic_programming_language), [weakly typed](https://en.wikipedia.org/wiki/Weak_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) and [multi-paradigm](https://en.wikipedia.org/wiki/Multi-paradigm_programming_language).

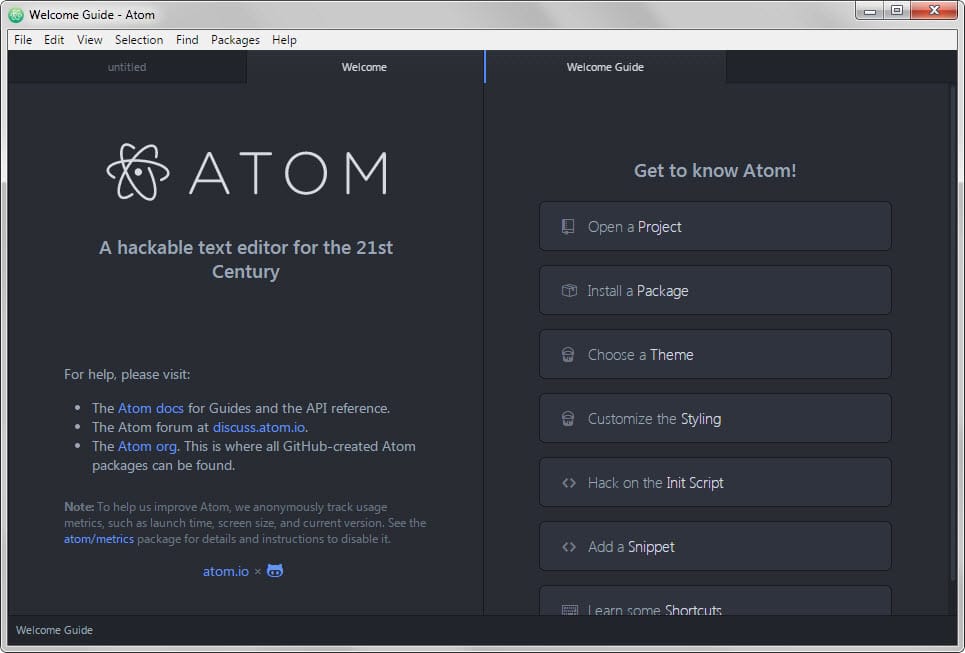
NodeJS: is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [run-time environment](https://en.wikipedia.org/wiki/Runtime_system) that executes JavaScript code [server-side](https://en.wikipedia.org/wiki/Server-side)

### IDE:

Visual Studio Community: is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) from [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It is used to develop [computer programs](https://en.wikipedia.org/wiki/Computer_program), as well as [web sites](https://en.wikipedia.org/wiki/Web_site), [web apps](https://en.wikipedia.org/wiki/Web_app), [web services](https://en.wikipedia.org/wiki/Web_service) and [mobile apps](https://en.wikipedia.org/wiki/Mobile_app). Visual Studio uses Microsoft software development platforms such as [Windows API](https://en.wikipedia.org/wiki/Windows_API), [Windows Forms](https://en.wikipedia.org/wiki/Windows_Forms), [Windows Presentation Foundation](https://en.wikipedia.org/wiki/Windows_Presentation_Foundation), [Windows Store](https://en.wikipedia.org/wiki/Windows_Store) and [Microsoft Silverlight](https://en.wikipedia.org/wiki/Microsoft_Silverlight). It can produce both [native code](https://en.wikipedia.org/wiki/Native_code) and [managed code](https://en.wikipedia.org/wiki/Managed_code).

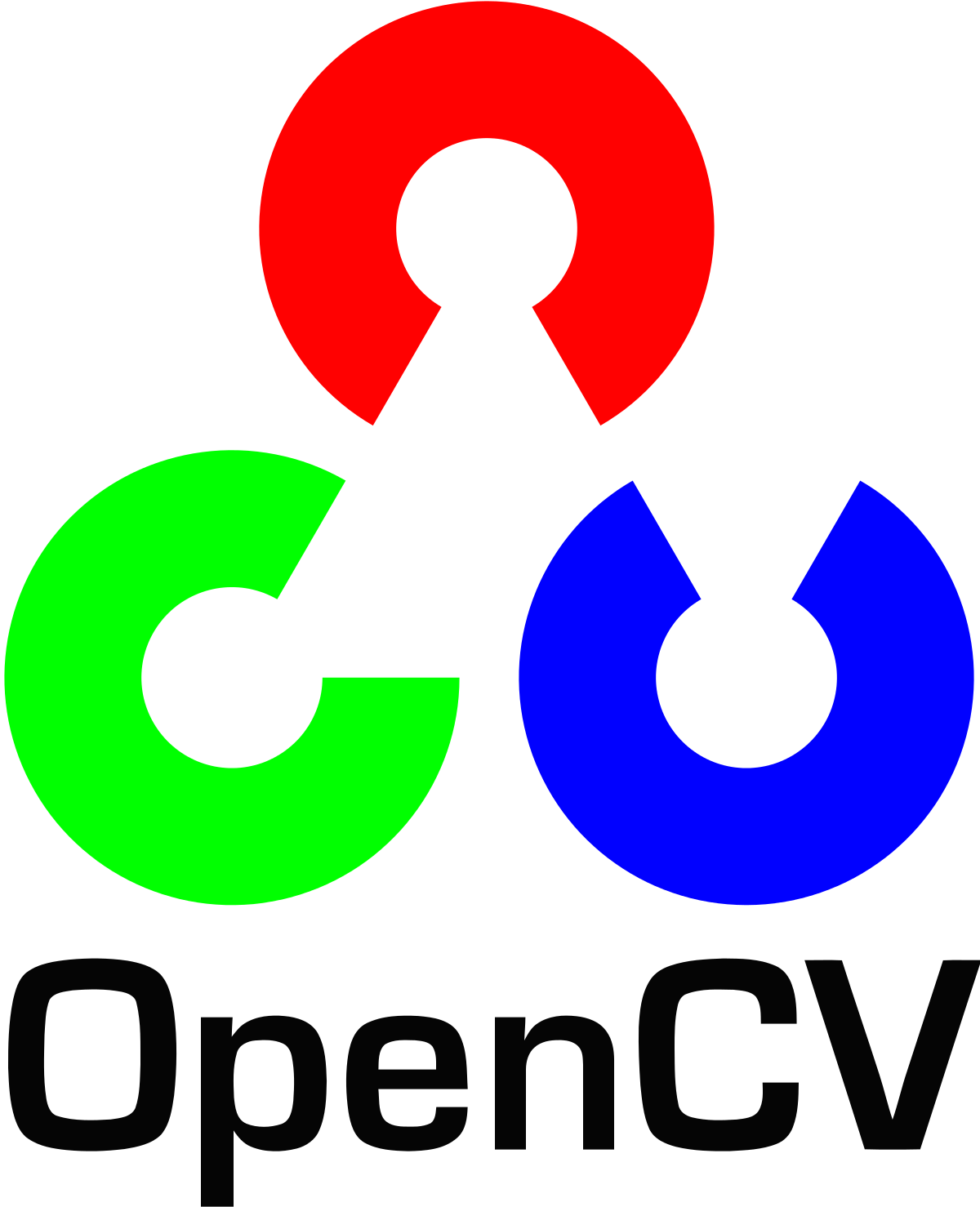




Atom: Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in Node.js, and embedded Git Control, developed by GitHub.

### Frameworks:

OpenCv: (Open Source Computer Vision) is a [library of programming functions](https://en.wikipedia.org/wiki/Library_(computing)) mainly aimed at real-time [computer vision](https://en.wikipedia.org/wiki/Computer_vision). Originally developed by [Intel](https://en.wikipedia.org/wiki/Intel_Corporation), it was later supported by [Willow Garage](https://en.wikipedia.org/wiki/Willow_Garage) then Itseez (which was later acquired by Intel). The library is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) and free for use under the [open-source](https://en.wikipedia.org/wiki/Open-source) [BSD license](https://en.wikipedia.org/wiki/BSD_license).



Opencv website: <https://opencv.org/>

Electron: is an open-source framework created and maintained by Github. It allows for the development of desktop GUI applications using front and back end components originally developed for web applications: Node.js runtime for the backend and Chromium for the frontend. Electron is the main GUI framework behind several notable open-source projects including GitHub's Atom and Microsoft's Visual Studio Code source code editors, the Tidal music streaming service desktop application and the Light Table IDE, in addition to the freeware desktop client for the Discord chat service.

### SDK:

Kinect SDK: The Kinect for Windows Software Development Kit (SDK) 2.0 enables developers to create applications that support gesture and voice recognition, using Kinect sensor technology on computers running Windows 8, Windows 8.1, and Windows Embedded Standard 8.



## Algorithm Implementation

### Input:

Input is an image or video(Frame). If the input is Image we process it in current stage,

If the input is video we must take it as sequence of frames.

But or input is live camera of Kinect.

Kinect provide more than one Input:

* **RGB IMAGE**
* **IR IMAGE**
* **DEPTH IMAGE**

We will take input as **RGB IMAGE** and 1920\*1700 resolution **1280x1024.**



Figure 1:code of input(RGB IMAGE) from Kinect

### Process

After taking the RGB image from kinect

* Detect the hand.
* Detect the shape of hand.
* Create Convex hull of the shape.
* Feature Extraction.
* Define the meaning of image.
* Return the word of meaning

#### Steps of process:

Detect the hand:After Analysis and some research we found two approaches to detect the hand

**First:** Detect the hand by shape and movement by Machine learning Algorithms to define the hand.

**Second:** Detect the hand by Color Range.

We make comparison between the two approaches to select the best of them.

|  |  |  |
| --- | --- | --- |
|  | Detect the hand by shape and movement | Detect the hand by Color Range. |
| Speed | slow | Fast |
| Complexity | Very complex | Less complex |
| Implementation | Hard to implement | Easy to implement |
| Accuracy | High | Middle(can improve) |

So we select the second approach to detect the hand. **Steps**:

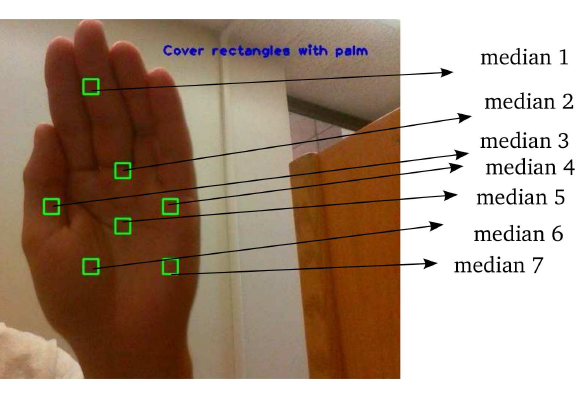
* Convert the Frame from RGB to HSV
* Get all Pixels of in Image between Ranges of Skin Color. 
* Get Shape of hand



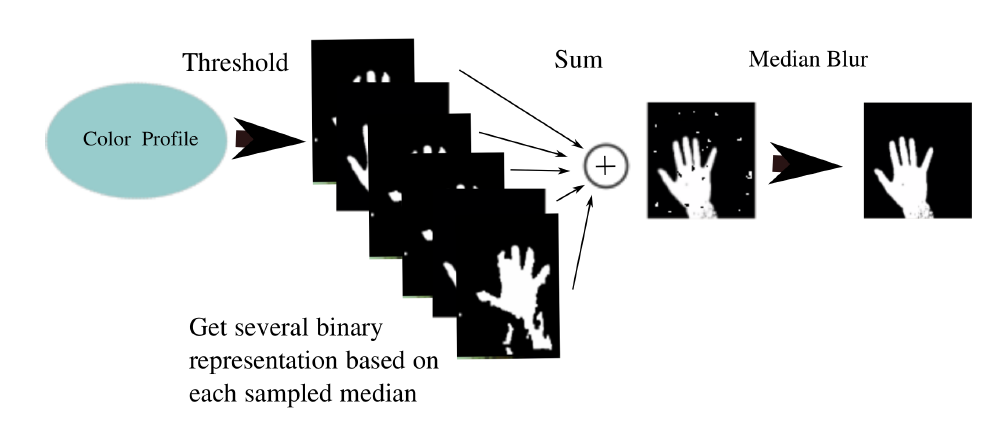
We have faced other problem in detect the hand the problem is that we have different colors of Skin. 

So we found new solution using the color, we can define the color range of hand in picture

By some tricks like we define rectangles to in area of hand to define the color of hand, cover the rectangles by your hand, then we select the median of rectangle to be average color.

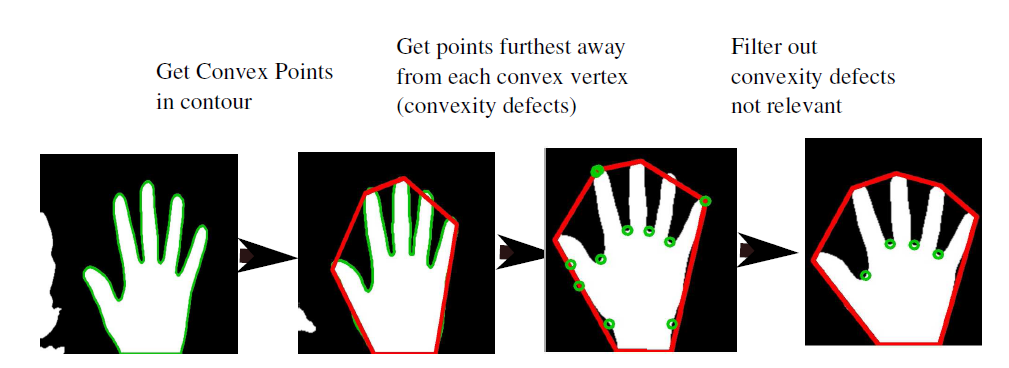


Aftter we select the color range can produce 7 image of shape of hand. we make image processing to remove noises and make the output be more than quality





After that we can make it shape and use some mathematic Convex Hell



**Convex Hull**:

the **convex hull** or **convex envelope** or **convex closure** of a set *X* of points in the Euclidean plane or in a Euclidean space(or, more generally, in an affine space over the reals) is the smallest convex set that contains *X*.

### Output

After Getting the shape we can detect the meaning of the shape by some tricks the computer vision will produce to us text of meaning the shape. So we can convert it to speech by using some tricks

Google API Translator:

Google provide text to speech by their API we used it achieve it, the tricks we use can be shown as

Open this Link:

[https://translate.google.com/translate\_tts?ie=UTF-8&q=Hello&tl=en](https://translate.google.com/translate_tts?ie=UTF-8&q=Hello&tl=en&client=tw-ob)

you will be could hear “Hello” by Female voice the tricks we will use this link to be achieved.

The Link Contain parameters:

* Host: google.com
* Sub-domain: Translate
* ie: Mean the decode and encode of language (UTF-8)
* q: this is word that will be speech.(Hello)
* tl: the language of word

we can make HTTP-request to download the voice and play it Showing as:

Download the voice:



To use **URLDownloadToFile** Function you should put before main this lines



To Play Sound



To use **mciSendString** function you should put this before main:



## NtKinect Framework:

We using NtKinect Framework to deal with Kinect “Wrapper” for easily document and memory management.

NtKinect can be easily use with Opencv:

Most important functions

**Open():** function to detect Kinect and open it

**Close():** function to close Kinect.

**setRGB():** Functiont to set mod of kinect to detect by color camera.

**setDepth():** Function to set mod of Kinect to detect by depth camera.

**setInfrared():** Function to set mod of Kinect to detect by Infrared sensor.

**setSkeleton():** Function to set mod of Kinect to detect the body joints.

Most Important Variables:

**rgbImage**: cv::Mat type and return color Image.

**depthImage**: CV::Mat type and return depth Image.

**infraredImage**: CV::Mat type and return infrared Image

**Skeleton**: vector<vector<Joint>> type and return body joints.