

PERTEMUAN II dan III

A. Persiapan Perangkat

a. Install Driver Wifi pada PC

Jika wifi sudah terinstal driver wifi langkah ini diabaikan. Lakukan penginstallan dengan benar dan pastikan driver wifi sudah terinstal dengan membuka Control Panel\Network and Internet\Network Connections apakah adapter wifi sudah muncul.

b. Setting PC sebagai Access Point win 7

Buka Command Prompt dengan mode access Administrator, kemudian ketikkan perintah

```
C:\Windows\system32>netsh wlan set hostednetwork mode=allow ssid=123 key=1234567890
```

Kemudian tekan enter, untuk mengaktifkan access point ketikkan perintah

```
C:\Windows\system32>netsh wlan start hostednetwork
```

Kemudian tekan enter, sedangkan untuk melihat status access point ketikkan perintah

```
C:\Windows\system32>netsh wlan show hostednetwork
```

Kemudian pastikan SmartPhone terkoneksi dengan PC melalui ssid yang telah dibuat.

c. Setting SmartPhone Android Sebagai WebCam

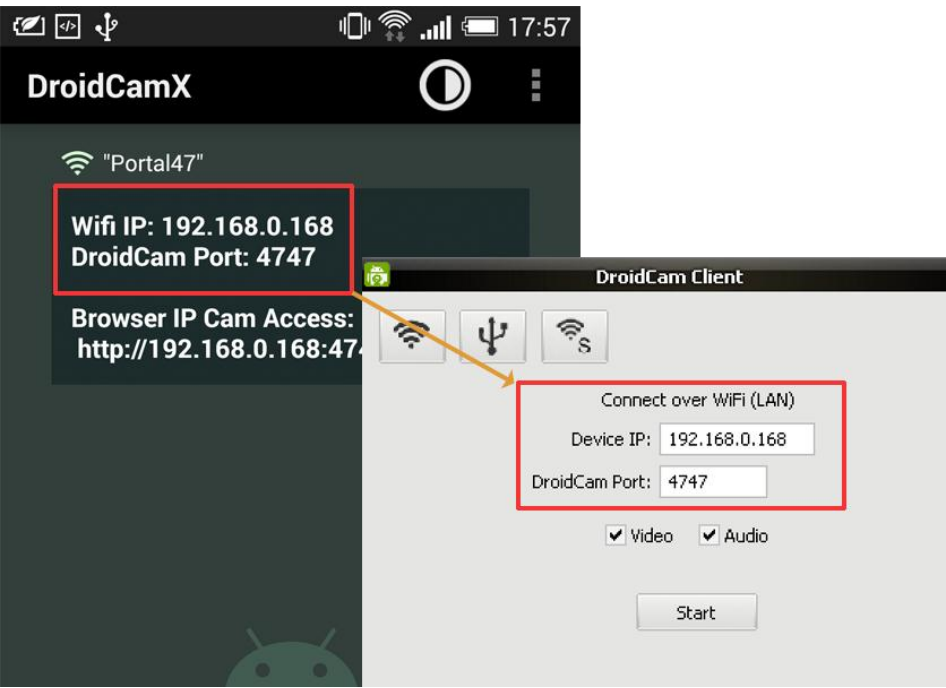
Install Aplikasi DroidCam / DroidCamX Wirelss Webcam, silahkan download di link ini <https://www.dev47apps.com/>,

d. Setting PC Client

Install Aplikasi PC Client untuk DroidCam, silahkan download di link ini <https://www.dev47apps.com/>,

e. Menghubungkan SmartPhone Camera dengan PC

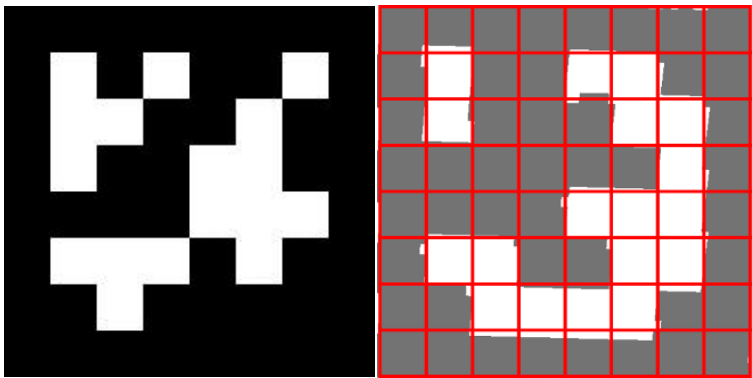
Buka aplikasi droidcam yang berada di PC (Droidcam Client) dan yang berada di SmartPhone, kemudian gunakan Connect over WiFi (LAN). Masukkan alamat ip yang tertera pada smartphone kedalam Device IP, dan kemudian tekan start.



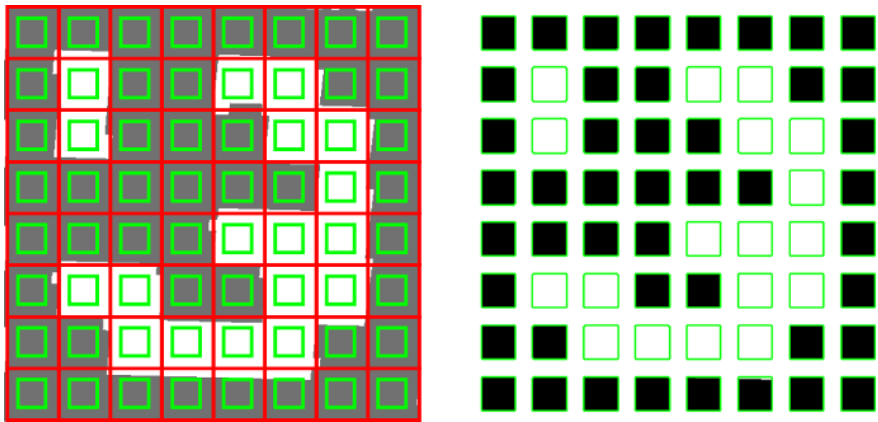
Droidcam for PC Client

f. Setting Square Marker

Seperti materi yang telah dijelaskan dalam pertemuan sebelumnya, bahwa dalam teknologi *augmented reality* salah satu input sensor yang dapat digunakan adalah pattern. Dasar menggunakan pattern yang digunakan sebagai marker adalah warna *black and White*, marker dengan basis pattern akan dihasilkan dari **marker generator**.



Generated Marker



Marker Call Margins

Aplikasi *augmented reality* dapat dikembangkan dengan menggunakan *library*, salah satunya adalah *jsaruco* yang dasar pengembangannya dari *arco* dengan basis bahasa pemrogramannya adalah C++, sedangkan *jsaruco* menggunakan dasar pengembangan bahasa pemrogramannya menggunakan *javascript*. Dengan menggunakan *jsaruco* aplikasi *augmented reality* yang dikembangkan dapat berjalan pada multi platform, karena berbasis web client. Untuk melakukan generated marker dapat kunjungi link berikut <https://terpconnect.umd.edu/~jwelsh12/enes100/markergen.html>.

B. Persiapan Library JSAruco

Pastikan webcam sudah dapat digunakan dengan baik dan bahan-bahan *library* sudah didownload, kemudian ikuti langkah-langkah sebagai berikut:

1. Masukkan file *aruco.js* dan *cv.js* kedalam satu folder dengan nama "latihan1"
2. Kemudian buka aplikasi *sublime text* arahkan *project* kedalam folder "latihan1"
3. Buatlah file *html* dengan nama *getwebcam.html*, dan simpan kedalam folder "latihan1" dan tuliskan

```
<html>
<head>
  <title>Latihan Augmented Reality Marker Detector</title>
  <script type="text/javascript" src="cv.js"></script> //Open Computer Vision Library
  <script type="text/javascript" src="aruco.js"></script> //aruco javascript library
</head>
<body style="font-family: monospace;">
  <center>
    <div style="margin: 10px;"><strong>-- Augmented Reality Marker Detector --</strong></div>
    <video id="video" autoplay="true" style="display:none;"></video> //video html5
    <canvas id="canvas" style="width:640px; height:480px;"></canvas> //canvas html5
    <div style="margin: 15px;"><strong>Powered by <a href="http://code.google.com/p/js-aruco/">js-
aruco</a></strong></div>
  </center>
</body>
</html>
```

4. Kemudian di antara tag head <head>....</head> tambahkan tag <script></script>, kemudian tambahkan variable *video*, *canvas*, *context*, *imageData* dan *detecot* (perhatikan penamaan dehan huruf besar atau kecil). lihat sctipt dibawah ini

```
<head>
  <script>
    var video, canvas, context, imageData, detector;
    //baris untuk menambahkan function
    window.onload = onLoad;
  </script>
</head>
```

5. Kemudian buat *function* dengan nama “*onload*” *function* ini berfungsi sebagai fungsi yang digunakan untuk mengambil media camera dengan memanfaatkan canvas. Lihat script di bawah ini

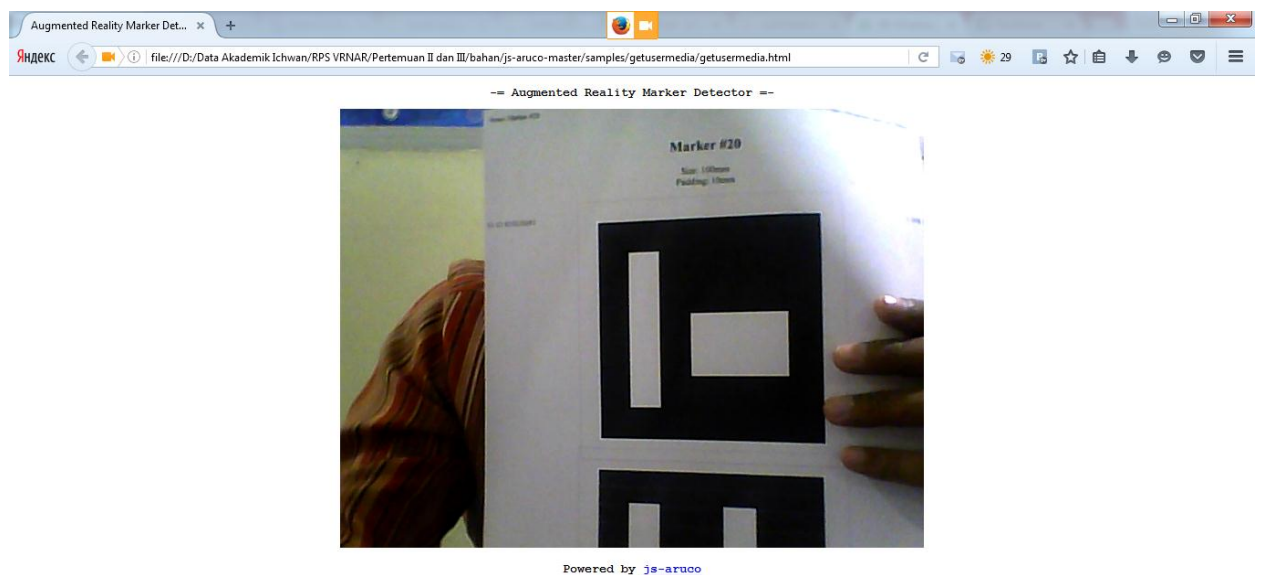
```
function onLoad(){
    video = document.getElementById("video");//mengambil element video dengan id "video"
    canvas = document.getElementById("canvas");//mengambil element canvas dengan id "canvas"
    context = canvas.getContext("2d");//menentukan konteks canvas
    canvas.width = parseInt(canvas.style.width);//mengambil lebar canvas dengan bil_bulat
    canvas.height = parseInt(canvas.style.height);//mengambil tinggi canvas dengan bil_bulat
    //untuk mengambil media device "webcam and microphone"
    if (navigator.mediaDevices === undefined) {
        navigator.mediaDevices = {};
    }
    if (navigator.mediaDevices.getUserMedia === undefined) {
        navigator.mediaDevices.getUserMedia = function(constraints) {
            var getUserMedia = navigator.webkitGetUserMedia || navigator.mozGetUserMedia;
            if (!getUserMedia) {
                return Promise.reject(new Error('getUserMedia is not implemented in this browser'));
            }
            return new Promise(function(resolve, reject) {
                getUserMedia.call(navigator, constraints, resolve, reject);
            });
        };
    }
    //setting media device
    navigator.mediaDevices
        .getUserMedia({ video: true })
        .then(function(stream) {
            if ("srcObject" in video) {
                video.srcObject = stream;
            } else {
                video.src = window.URL.createObjectURL(stream);
            }
        })
        .catch(function(err) {
            console.log(err.name + ": " + err.message);
        });
    detector = new AR.Detector();//digunakan untuk mengenali marker terhubung dengan jsaruco
    requestAnimationFrame(lat_ar);
}
```

6. Kemudian buatlah *function* dengan nama “*snapshot*” yang berfungsi sebagai pengambilan gambar streaming camera webcam. Lihat script di bawah ini

```
function snapshot(){
    context.drawImage(video, 0, 0, canvas.width, canvas.height);
    imageData = context.getImageData(0, 0, canvas.width, canvas.height);
}
```

7. Kemudian tambahkan *function* dengan nama “*lat_ar*”, *function* ini digunakan untuk memanggil marker yang terdeteksi. Lihat script di bawah ini

```
function lat_ar(){
    requestAnimationFrame(lat_ar);
    if (video.readyState === video.HAVE_ENOUGH_DATA){
        snapshot();
        var markers = detector.detect(imageData);
        drawinfo(markers);
    }
}
```



Webcam dapat berfungsi namun belum dapat mengenali marker yang tertangkap oleh camera.

8. Untuk menambahkan informasi tambahkan *function* dengan nama “drawinfo”. Ikuti script di bawah ini.

```
function drawinfo(markers){
    var sudut, i, j, x, y;
    for ( i = 0; i !== markers.length; ++ i){
        sudut = markers[i].corners;
        x = Infinity; y = Infinity;
        for (j = 0; j !== sudut.length; ++ j){
            corner = sudut[j];
            x = Math.min(x, corner.x);
            y = Math.min(y, corner.y);
        }
        context.font = "30px arial";
        context.fillStyle = "blue";
        context.fillText("Augmented Reality", x, y);
        var gambar = new Image();
        gambar.src = 'logo.png';
        context.drawImage(gambar, x, y);
    }
}
```