# SIDANG TUGAS AKHIR

## SISTEM PEMANTAU KEPADATAN LALU LINTAS KENDARAAN OTOMATIS DI JALAN RAYA DENGAN MENGGUNAKAN METODE *NORMALIZED SUM-SQUARE DIFFERENCES* (NSSD)

**(Studi kasus di PT. Jasa Marga (persero))**




**Asep Setiawan - 10108902**

**PROGRAM STUDI S1**
**JURUSAN TEKNIK INFORMATIKA**
**FAKULTAS TEKNIK DAN ILMU KOMPUTER**
**UNIVERSITAS KOMPUTER INDONESIA**


JASAMARGA
Indonesia Highway Corp.

**SISTEM PEMANTAU KEPADATAN LALU LINTAS KENDARAAN OTOMATIS DI JALAN RAYA DENGAN MENGGUNAKAN METODE *NORMALIZED SUM-SQUARE DIFFERENCES* (NSSD)**
**(Studi kasus di PT. Jasa Marga (persero))**

**Bandung, 6 Agustus 2012**

**Dosen Penguji I,**

**Andri Heryandi, S.T., M.T.**
**NIP. 41277006007**

**Dosen Penguji II,**

**Irawan Afrianto, S.T., M.T.**
**NIP. 41277006009**

**Dosen Penguji III,**

**Mira Kania Sabariah, S.T., M.T.**
**NIP. 41277006008**

UNIVERSITAS KOMPUTER INDONESIA
UNIKOM

JASAMARGA
Indonesia Highway Corp.

**SISTEM PEMANTAU KEPADATAN LALU LINTAS KENDARAAN OTOMATIS DI JALAN RAYA DENGAN MENGGUNAKAN METODE *NORMALIZED SUM-SQUARE DIFFERENCES* (NSSD)**
**(Studi kasus di PT. Jasa Marga (persero))**



1. Pendahuluan
2. Landasan Teori
3. Analisis dan Perancangan
4. Implementasi
5. Demo Aplikasi

UNIVERSITAS KOMPUTER INDONESIA
UNIKOM

JASAMARGA
Indonesia Highway Corp.

## LATAR BELAKANG MASALAH

# PT.Jasa Marga(persero)







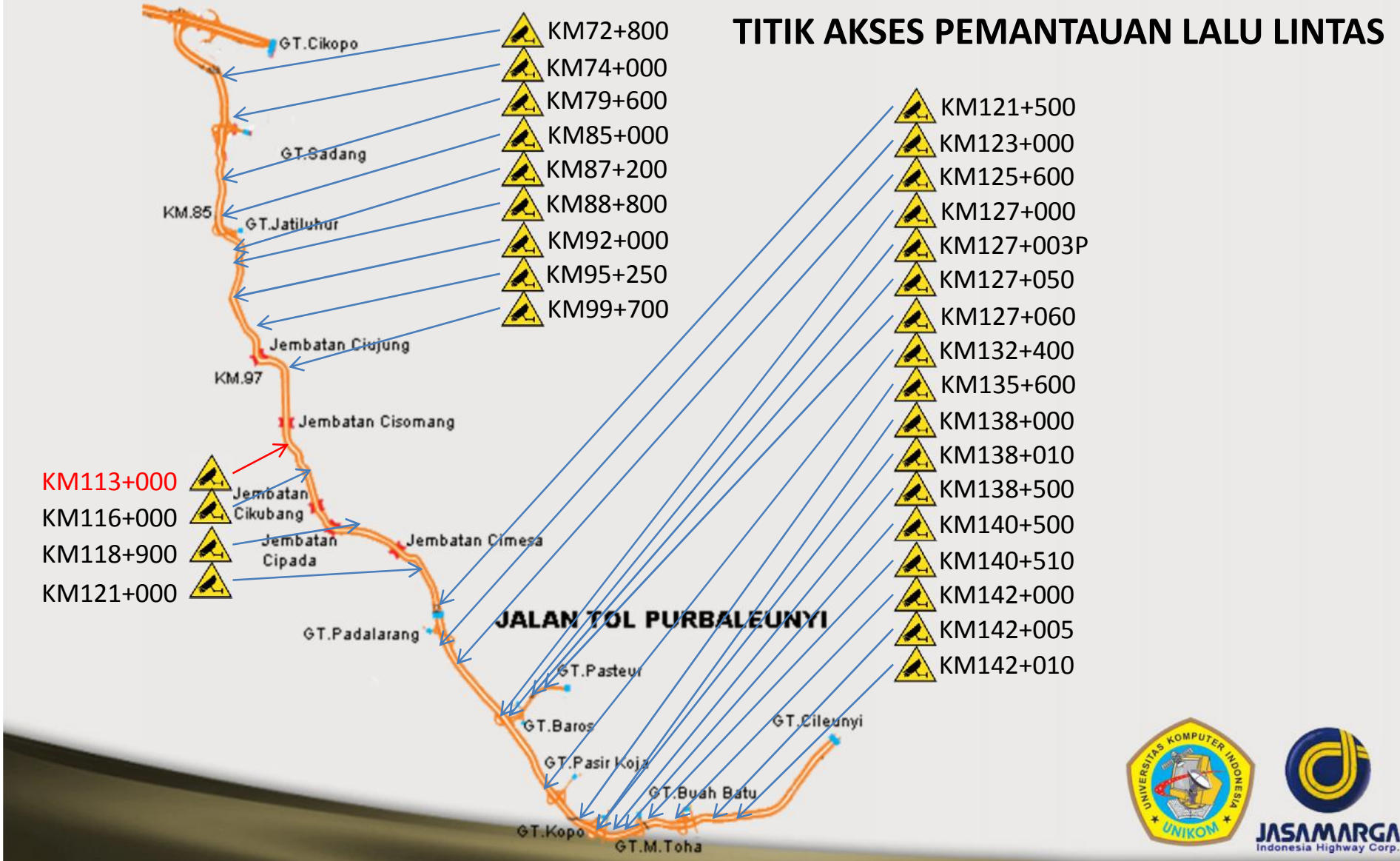



UNIVERSITAS KOMPUTER INDONESIA
UNIKOM

JASAMARGA
Indonesia Highway Corp.

# 1 Pendahuluan

## LATAR BELAKANG MASALAH



## LATAR BELAKANG MASALAH

# PEMANTAUAN DAN PERKIRAAN KEPADATAN LALU LINTAS



LANDASAN TEORI

**Metoda Normalize Sum-Square Differences(NSSD)**

Metoda ini digunakan sebagai sensor untuk mengenali objek. Yang mengambil selisih jumlah pixel frame dan background dan dikuadratkan.

**Perhitungan NSSD**

*Normalized Sum Squared Differences* (NSSD) yang terdapat pada setiap frame yang telah dikonversi menjadi *grayscale*. Perhitungan dilakukan di dalam ROI. Dalam perhitungan NSSD dibutuhkan nilai nilai piksel *background* dan nilai piksel frame.

## LANDASAN TEORI

**Langkah-langkah Metode NSSD adalah sebagai berikut:**

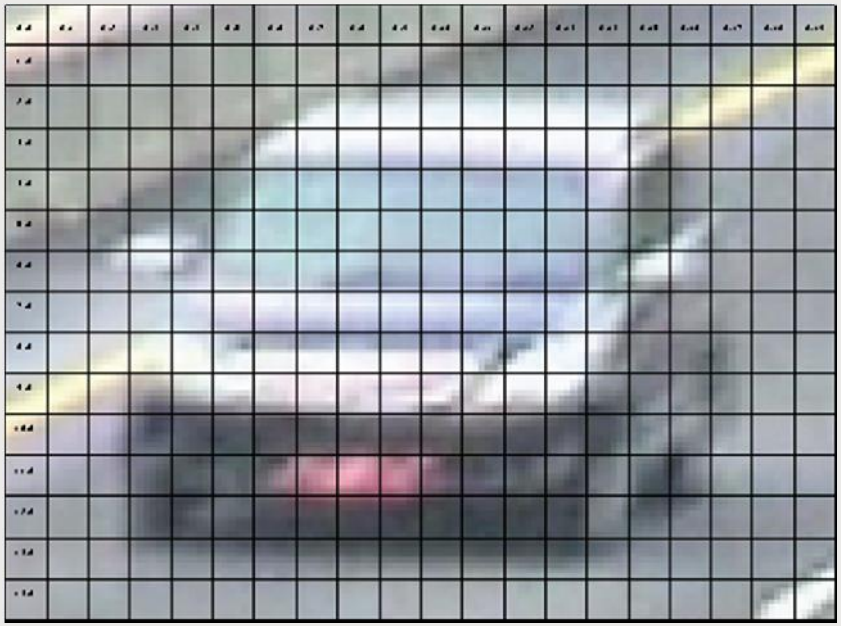
1. Citra *background*
   Tentukan citra *background* dan nilai pikselnya. Citra ini harus sudah dalam bentuk citra abu-abu.
2. Tentukan citra frame dan nilai pikselnya
   Tentukan citra *frame* dan nilai pikselnya. Citra ini juga harus sudah dalam bentuk citra abu-abu.
3. Untuk setiap frame, lakukan pemrosesan setiap piksel yang ada di dalam ROI menggunakan rumus NSSD berikut:

$$NSSD = \frac{\Sigma \left( P_f(i) - P_b(i) \right)^2}{N}$$

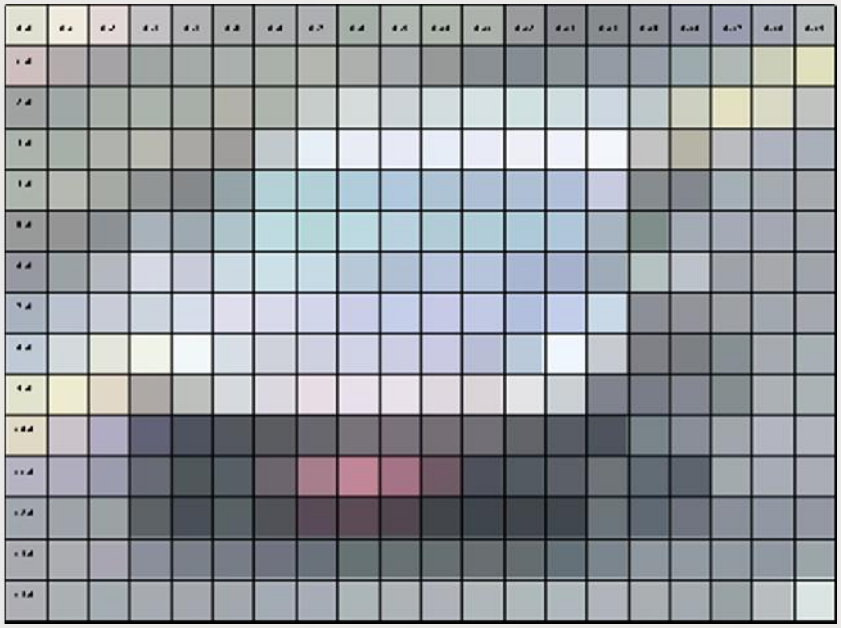
dengan $P_f$ adalah nilai piksel *frame*, $P_b$ adalah nilai piksel *background*, dan N adalah jumlah piksel yang diproses.

4. Tentukan nilai rata-rata NSSD.

## Contoh Penerapan Pemrosesan NSSD



Citra Berwarna RGB



Citra Berwarna RGB dalam piksel

## LANDASAN TEORI

| X / Y | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | (225,218,210) | (236,231,225) | (217,216,214) | (191,193,192) | (171,175,174) | (163,169,167) | (166,172,168) | (168,177,172) | (162,173,167) | (170,181,175) | (168,178,170) | (167,176,173) | (147,157,159) | (131,142,148) | (129,143,146) | (135,150,153) | (141,154,160) | (149,160,162) | (159,169,161) | (174,182,169) |
| 1 | (196,193,186) | (173,172,167) | (162,163,158) | (161,163,160) | (164,169,165) | (167,173,169) | (167,173,171) | (172,181,178) | (166,177,173) | (161,172,168) | (144,155,151) | (135,145,144) | (133,142,147) | (139,150,156) | (142,156,159) | (147,161,162) | (157,169,169) | (171,182,178) | (194,204,196) | (212,220,209) |
| 2 | (156,163,156) | (159,166,159) | (166,173,165) | (169,176,168) | (166,171,164) | (170,176,172) | (168,177,174) | (193,203,202) | (206,216,217) | (199,209,210) | (207,217,219) | (211,220,225) | (208,217,224) | (206,215,222) | (200,211,217) | (187,197,198) | (199,204,198) | (218,221,212) | (208,213,207) | (186,192,188) |
| 3 | (166,177,171) | (163,174,166) | (169,179,170) | (175,185,176) | (160,170,162) | (150,161,157) | (184,200,199) | (220,238,240) | (220,238,240) | (219,237,241) | (221,238,245) | (220,237,245) | (226,240,249) | (229,244,251) | (234,247,255) | (185,195,197) | (172,177,171) | (182,187,180) | (172,178,176) | (166,175,174) |
| 4 | (168,178,170) | (173,182,177) | (158,169,165) | (139,151,149) | (127,141,141) | (143,163,164) | (175,205,205) | (170,204,205) | (170,201,206) | (168,198,206) | (163,193,201) | (162,192,202) | (163,191,202) | (169,191,202) | (187,204,212) | (129,142,148) | (128,139,141) | (162,172,173) | (162,172,174) | (162,169,175) |
| 5 | (145,156,152) | (140,150,149) | (136,146,147) | (165,176,180) | (154,169,174) | (171,192,197) | (182,213,216) | (174,209,213) | (179,213,222) | (173,207,217) | (167,201,213) | (168,200,213) | (168,197,213) | (169,195,208) | (160,178,188) | (127,142,149) | (158,171,177) | (160,171,177) | (157,168,174) | (159,168,175) |
| 6 | (144,155,159) | (151,162,166) | (174,185,191) | (204,215,221) | (190,203,211) | (197,211,220) | (198,218,229) | (190,214,226) | (173,199,214) | (165,191,208) | (171,196,216) | (172,194,217) | (160,182,206) | (155,176,195) | (152,170,180) | (175,190,195) | (181,192,196) | (152,163,167) | (158,169,173) | (154,165,169) |
| 7 | (166,177,179) | (182,193,195) | (193,203,205) | (201,208,214) | (210,219,226) | (212,222,231) | (205,217,229) | (199,212,228) | (190,206,222) | (189,206,224) | (188,203,224) | (184,199,222) | (172,188,211) | (187,204,222) | (196,213,223) | (133,146,152) | (139,150,154) | (151,161,163) | (158,169,173) | (157,171,174) |
| 8 | (190,200,192) | (208,215,208) | (224,227,220) | (238,240,235) | (242,248,248) | (210,218,221) | (202,206,217) | (202,206,218) | (201,207,221) | (198,204,218) | (195,201,217) | (178,185,203) | (185,199,212) | (235,249,255) | (191,202,208) | (124,133,138) | (123,133,135) | (133,144,146) | (158,172,175) | (161,175,178) |
| 9 | (217,223,213) | (229,232,225) | (214,215,209) | (168,170,167) | (184,190,188) | (207,215,217) | (213,213,221) | (223,220,231) | (228,222,234) | (229,223,237) | (215,213,227) | (208,210,223) | (217,226,235) | (195,206,212) | (126,135,140) | (121,130,135) | (130,139,144) | (131,142,146) | (163,177,180) | (164,178,181) |
| 10 | (213,213,213) | (194,195,197) | (170,173,180) | (104,108,117) | (87,96,101) | (91,98,104) | (102,99,108) | (118,106,118) | (135,114,129) | (134,116,130) | (124,113,127) | (118,116,127) | (103,107,118) | (94,103,110) | (87,96,101) | (124,135,139) | (134,145,149) | (155,166,172) | (171,182,188) | (171,182,188) |
| 11 | (180,179,184) | (172,173,178) | (154,158,169) | (106,114,125) | (87,97,106) | (96,105,110) | (111,110,116) | (147,137,146) | (168,148,160) | (147,129,141) | (109,101,112) | (91,91,99) | (92,99,107) | (96,105,112) | (111,122,126) | (102,116,119) | (97,110,116) | (156,169,175) | (162,173,179) | (163,174,180) |
| 12 | (161,166,170) | (157,162,168) | (152,159,167) | (98,107,114) | (81,92,98) | (95,106,112) | (89,94,100) | (92,91,97) | (96,91,98) | (89,86,93) | (79,82,89) | (74,83,88) | (74,85,89) | (72,86,89) | (109,124,127) | (99,114,117) | (111,124,130) | (133,146,152) | (140,153,159) | (140,153,159) |
| 13 | (162,170,173) | (165,172,178) | (160,167,173) | (136,145,150) | (122,133,139) | (119,130,134) | (113,122,127) | (111,119,122) | (112,117,121) | (112,117,121) | (107,116,121) | (107,118,122) | (103,117,118) | (106,120,121) | (121,136,139) | (138,153,156) | (142,156,159) | (142,156,159) | (140,154,157) | (151,165,168) |
| 14 | (167,175,178) | (166,174,177) | (162,171,176) | (162,171,176) | (159,168,173) | (158,167,172) | (161,172,176) | (161,172,176) | (168,178,180) | (168,178,180) | (169,179,181) | (172,182,184) | (170,181,183) | (170,181,183) | (167,181,184) | (161,175,178) | (158,172,175) | (148,162,163) | (177,189,189) | (212,224,224) |

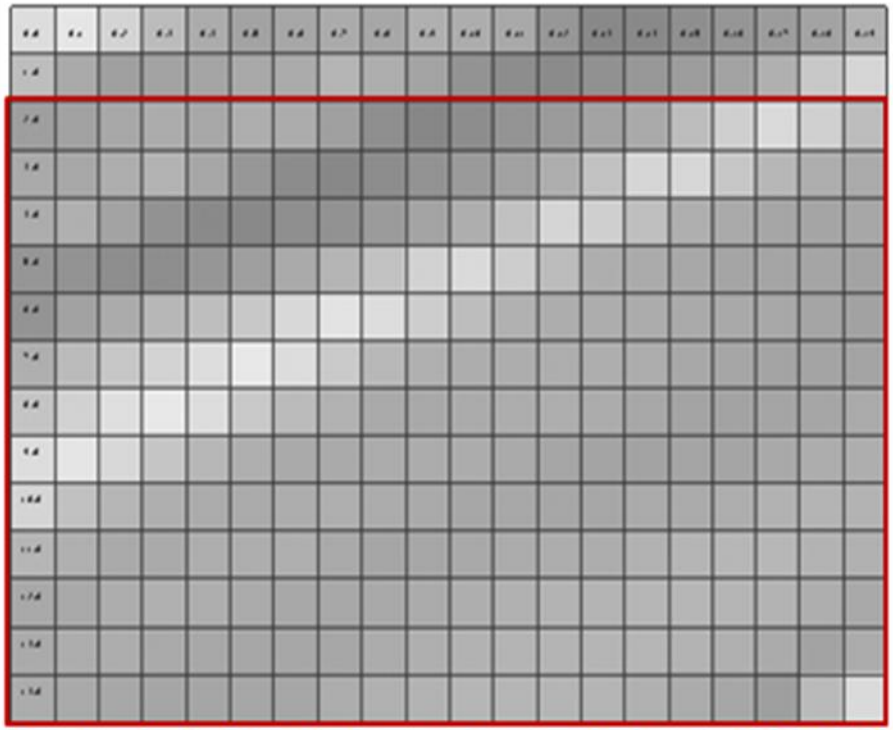
**Nilai Piksel Citra RGB**

## LANDASAN TEORI

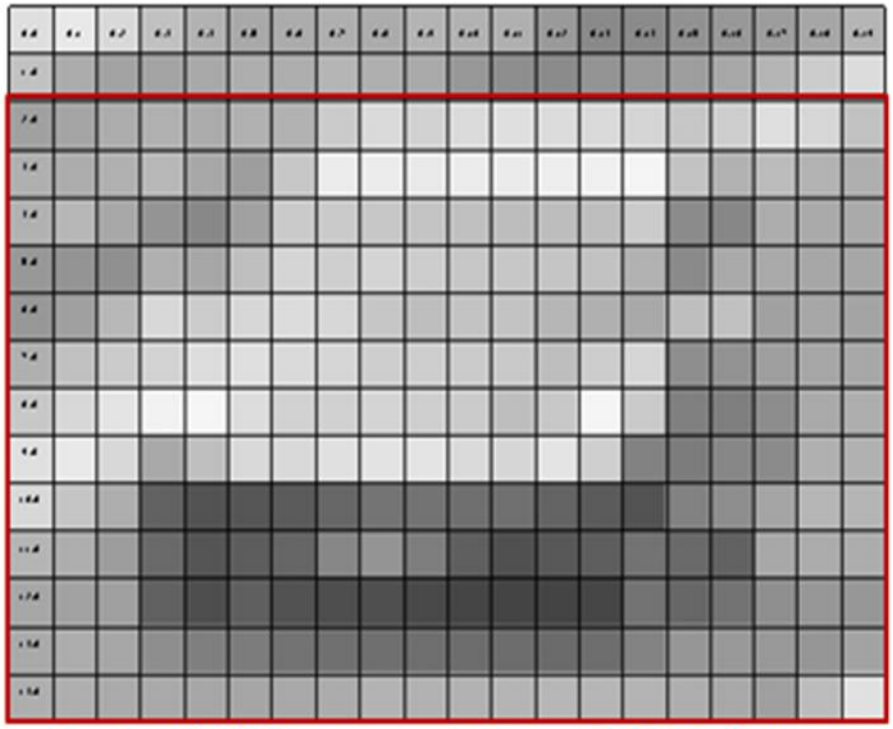
| X/Y | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 218 | 231 | 216 | 192 | 174 | 167 | 170 | 174 | 170 | 178 | 175 | 173 | 155 | 140 | 140 | 147 | 151 | 157 | 166 | 179 |
| 1 | 193 | 171 | 162 | 162 | 167 | 171 | 171 | 178 | 174 | 169 | 152 | 142 | 140 | 148 | 153 | 158 | 166 | 179 | 201 | 217 |
| 2 | 161 | 164 | 170 | 173 | 169 | 174 | 174 | 200 | 213 | 206 | 215 | 218 | 215 | 213 | 209 | 194 | 202 | 219 | 211 | 190 |
| 3 | 174 | 171 | 176 | 182 | 167 | 158 | 196 | 234 | 234 | 233 | 234 | 233 | 237 | 241 | 244 | 193 | 175 | 185 | 176 | 173 |
| 4 | 175 | 179 | 166 | 148 | 138 | 158 | 198 | 196 | 194 | 192 | 187 | 186 | 185 | 187 | 200 | 139 | 136 | 169 | 170 | 167 |
| 5 | 153 | 147 | 143 | 173 | 166 | 187 | 206 | 201 | 206 | 200 | 194 | 194 | 191 | 190 | 174 | 139 | 168 | 169 | 166 | 166 |
| 6 | 152 | 159 | 183 | 213 | 200 | 208 | 214 | 209 | 194 | 186 | 192 | 190 | 179 | 172 | 166 | 187 | 189 | 160 | 166 | 162 |
| 7 | 174 | 190 | 201 | 206 | 217 | 220 | 215 | 210 | 203 | 203 | 201 | 197 | 186 | 201 | 210 | 143 | 147 | 159 | 166 | 168 |
| 8 | 197 | 213 | 225 | 239 | 246 | 216 | 205 | 206 | 206 | 203 | 200 | 184 | 196 | 246 | 200 | 131 | 131 | 141 | 169 | 172 |
| 9 | 221 | 230 | 214 | 169 | 188 | 213 | 213 | 221 | 224 | 225 | 214 | 210 | 224 | 204 | 133 | 128 | 137 | 139 | 174 | 175 |
| 10 | 213 | 194 | 172 | 107 | 94 | 96 | 100 | 109 | 119 | 120 | 116 | 117 | 106 | 101 | 94 | 132 | 142 | 164 | 180 | 180 |
| 11 | 179 | 173 | 157 | 113 | 95 | 103 | 110 | 139 | 153 | 133 | 103 | 91 | 98 | 103 | 119 | 113 | 107 | 166 | 171 | 172 |
| 12 | 165 | 161 | 158 | 105 | 90 | 104 | 93 | 91 | 92 | 87 | 81 | 81 | 82 | 83 | 121 | 111 | 121 | 143 | 150 | 150 |
| 13 | 168 | 170 | 165 | 143 | 131 | 127 | 120 | 117 | 116 | 116 | 114 | 115 | 114 | 117 | 133 | 150 | 153 | 153 | 151 | 162 |
| 14 | 173 | 172 | 169 | 169 | 166 | 165 | 169 | 169 | 176 | 176 | 177 | 180 | 178 | 178 | 178 | 172 | 169 | 159 | 186 | 221 |

**Nilai Piksel Citra *gray scale***

## LANDASAN TEORI



Gambar II.31 Citra background grayscale dengan ROI 13x20 piksel



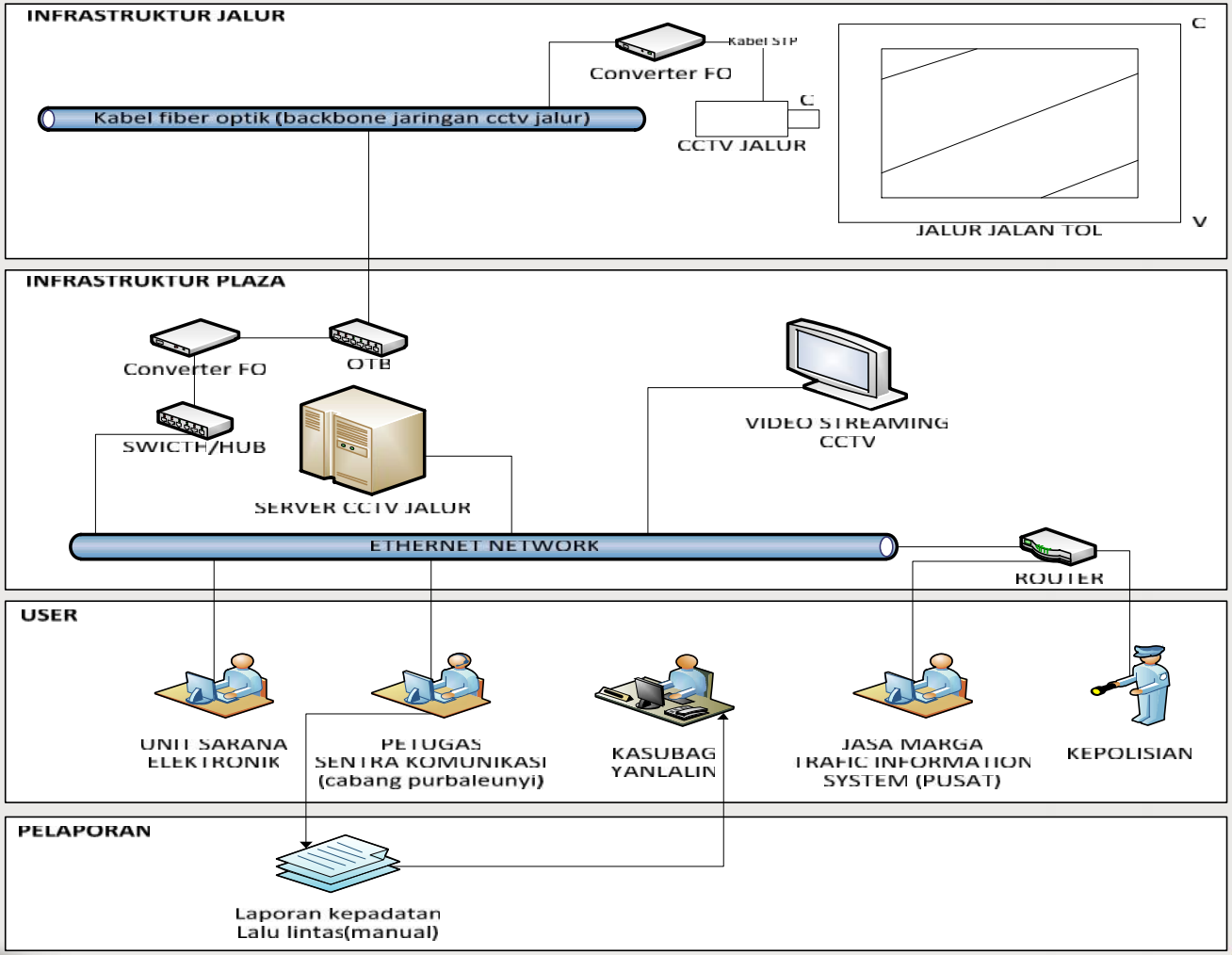
Gambar II.32 Citra grayscale dengan ROI 13x20 piksel

## LANDASAN TEORI

| X/Y | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 4 | 4 | 9 | 1 | 1 | 0 | 4 | 1521 | 4900 | 5041 | 5476 | 4900 | 3481 | 2401 | 1444 | 36 | 9 | 4 | 4 | 0 |
| **1** | 25 | 16 | 9 | 9 | 0 | 49 | 3136 | 9409 | 8649 | 7396 | 6400 | 5184 | 3969 | 2116 | 900 | 484 | 676 | 4 | 9 | 9 |
| **2** | 4 | 16 | 9 | 1 | 1 | 441 | 3025 | 2401 | 1521 | 841 | 144 | 64 | 729 | 400 | 100 | 1369 | 1225 | 16 | 9 | 1 |
| **3** | 9 | 0 | 4 | 961 | 225 | 676 | 1444 | 400 | 144 | 81 | 576 | 121 | 9 | 225 | 4 | 841 | 9 | 16 | 0 | 4 |
| **4** | 16 | 25 | 144 | 961 | 100 | 49 | 9 | 289 | 729 | 361 | 4 | 144 | 25 | 1 | 36 | 289 | 400 | 64 | 1 | 1 |
| **5** | 0 | 9 | 4 | 16 | 9 | 144 | 36 | 64 | 324 | 729 | 784 | 576 | 169 | 729 | 1369 | 576 | 324 | 25 | 4 | 16 |
| **6** | 0 | 9 | 4 | 36 | 625 | 225 | 324 | 784 | 1296 | 1156 | 784 | 100 | 441 | 5476 | 1024 | 1089 | 1089 | 576 | 9 | 1 |
| **7** | 0 | 1 | 9 | 784 | 16 | 1444 | 1849 | 2500 | 2704 | 2704 | 1764 | 1600 | 3364 | 1600 | 961 | 1369 | 784 | 900 | 4 | 1 |
| **8** | 1 | 0 | 100 | 4624 | 5929 | 5625 | 5329 | 4356 | 2809 | 2500 | 2601 | 2500 | 3721 | 4489 | 5776 | 1521 | 1024 | 196 | 1 | 1 |
| **9** | 9 | 9 | 169 | 3249 | 6400 | 5184 | 4096 | 1024 | 169 | 1225 | 4225 | 6400 | 5625 | 4900 | 3481 | 4624 | 5776 | 256 | 100 | 25 |
| **10** | 25 | 49 | 225 | 4900 | 7056 | 4356 | 5929 | 5776 | 6084 | 7225 | 8464 | 9025 | 9604 | 9216 | 3844 | 5041 | 3600 | 1296 | 576 | 361 |
| **11** | 9 | 16 | 64 | 729 | 1369 | 1681 | 2304 | 2601 | 3136 | 3600 | 4225 | 4225 | 4624 | 4225 | 2500 | 961 | 529 | 361 | 144 | 100 |
| **12** | 1 | 4 | 1 | 25 | 1 | 4 | 4 | 16 | 0 | 9 | 25 | 4 | 16 | 9 | 1 | 1 | 1 | 1 | 1 | 9 |

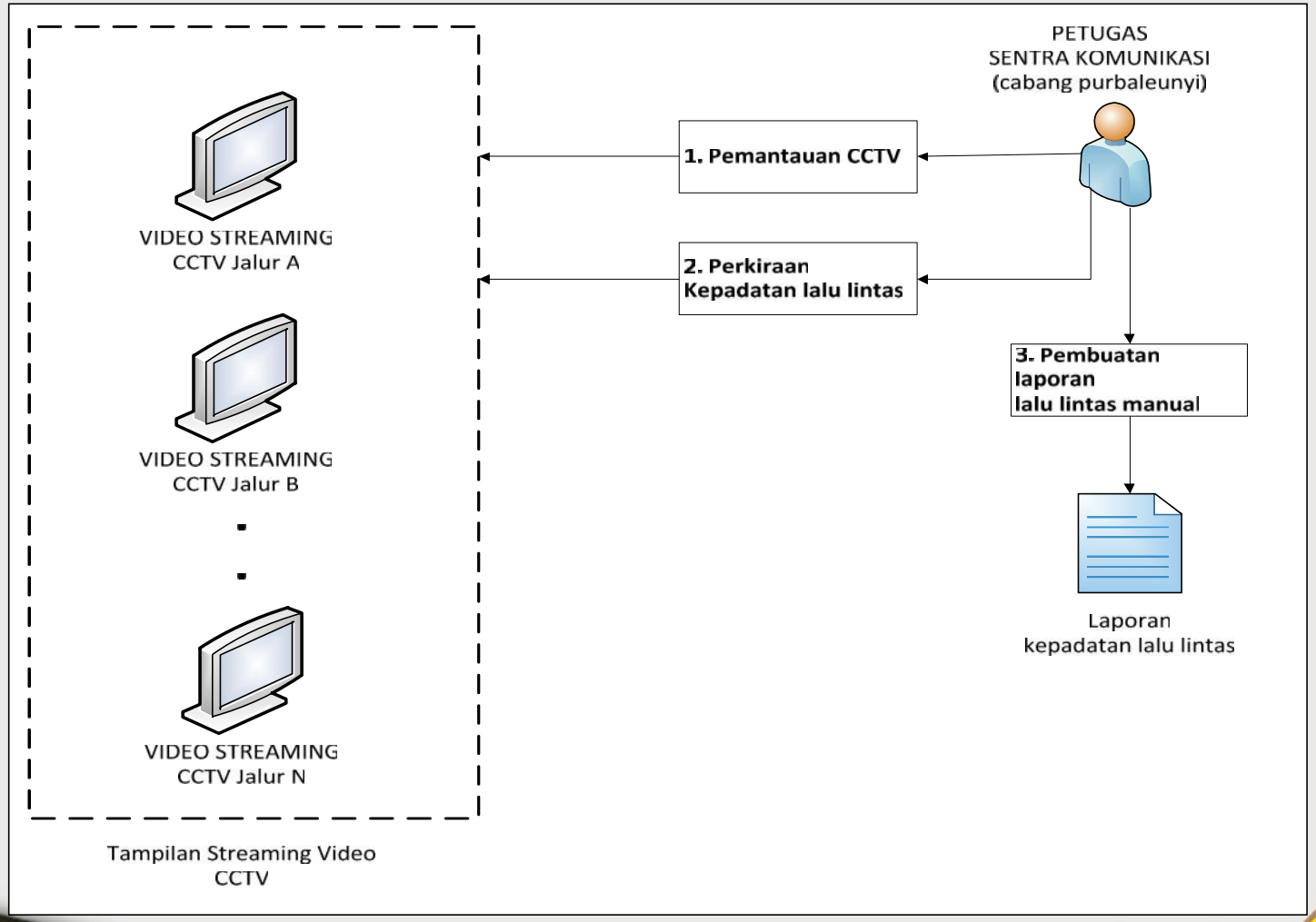
**Nilai Piksel *(Frame-Backgound)*$^2$ pada frame**

## ANALISIS SISTEM



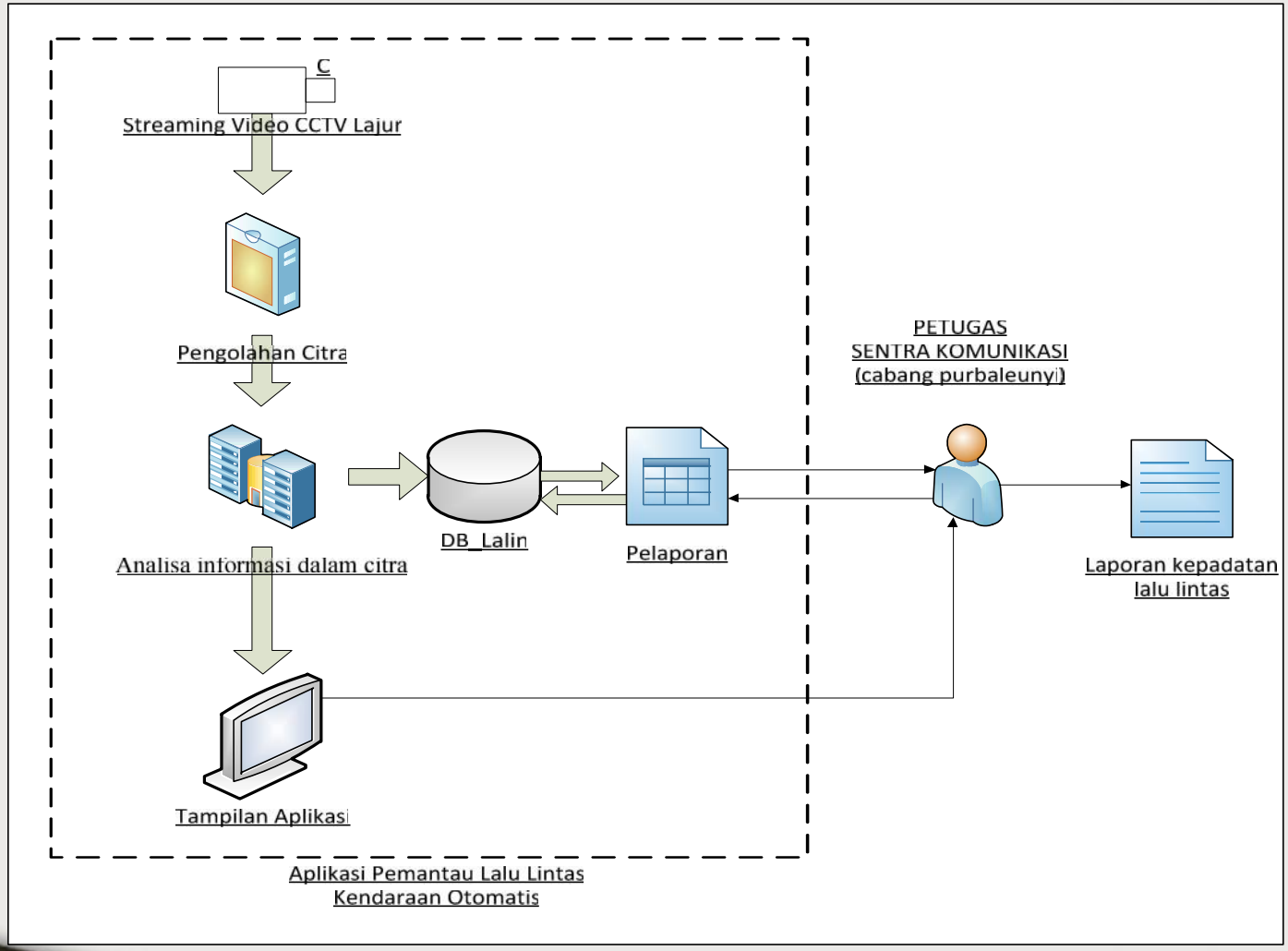

**Arsitektur Sistem Yang Sedang Berjalan**

## ANALISIS MASALAH




**Deskripsi Sistem Yang Sedang Berjalan**

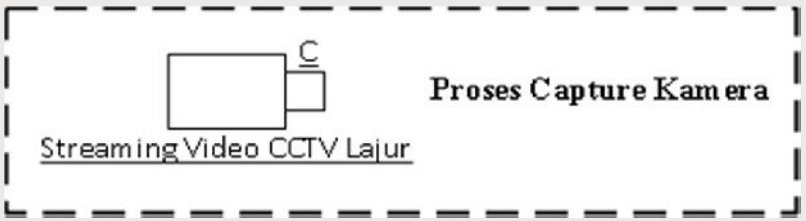
## ARSITEKTUR SISTEM YANG AKAN DIBANGUN




**Deskripsi Sistem Yang Akan Dibangun**

## ANALISIS PROSES PENGHITUNGAN KENDARAAN

1. Proses Capture Kamera




2. Penentuan Region of Interest
3. Proses Grayscaling
4. Proses NSSD
5. Thresholding

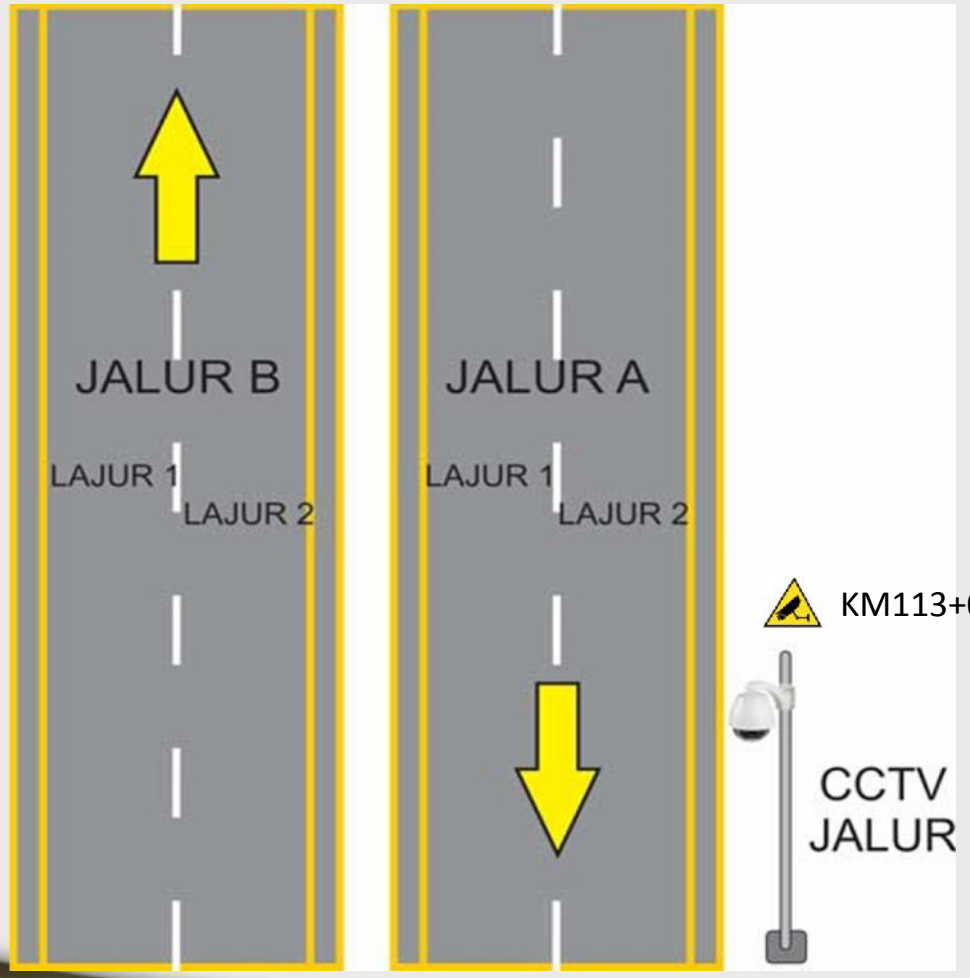



6. Penghitungan Kendaraan dan Tingkat Kepadatan



ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 1. Proses Capture Kamera



## ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 1. Proses Capture Kamera

```
Procedure VideoCapture(mjpegSource : MJPEGStream)

Kamus
   height, width : integer
   URL : array[] of character
   lastFrame : Bitmap

struct MJPEGStream
   procedure VideoSource(URL : array[] of character)
   procedure Start()
   procedure Stop()
   procedure Bitmap()

Algoritma
   URL ← "http://202.90.195.131:218/axis-cgi/mjpg/video.cgi"
   mjpegSource.VideoSource(URL)
   mjpegSource.Start()
   if (mjpegSource <> NULL) then
      lastFrame ← mjpegSource.Bitmap
      height ← lastFrame.Height
      width ← lastFrame.Width
   EndIf
   mjpegSource.Stop()
```

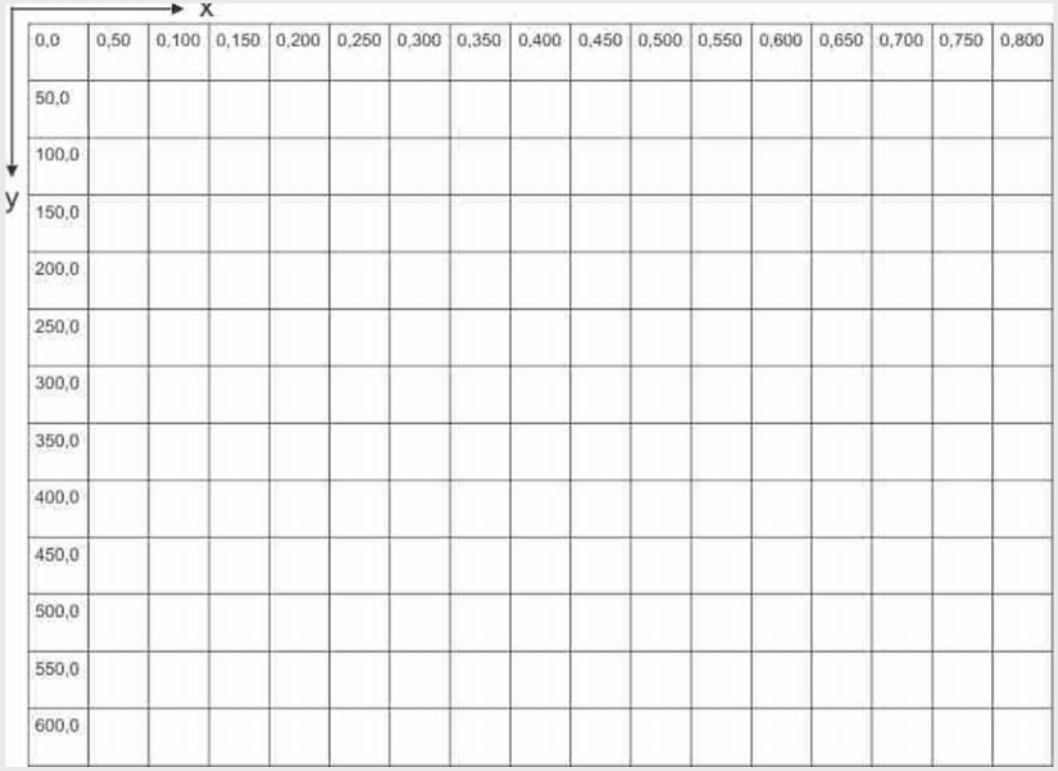



**Hasil *Capture* Kamera**

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 2. PENENTUAN *REGION OF INTEREST*

## 2.1 Tentukan dimensi citra




**Jumlah piksel dalam *Region of Interest* (800x600)**

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 2. PENENTUAN *REGION OF INTEREST*

## 2.2 Tentukan titik-titik daerah objek

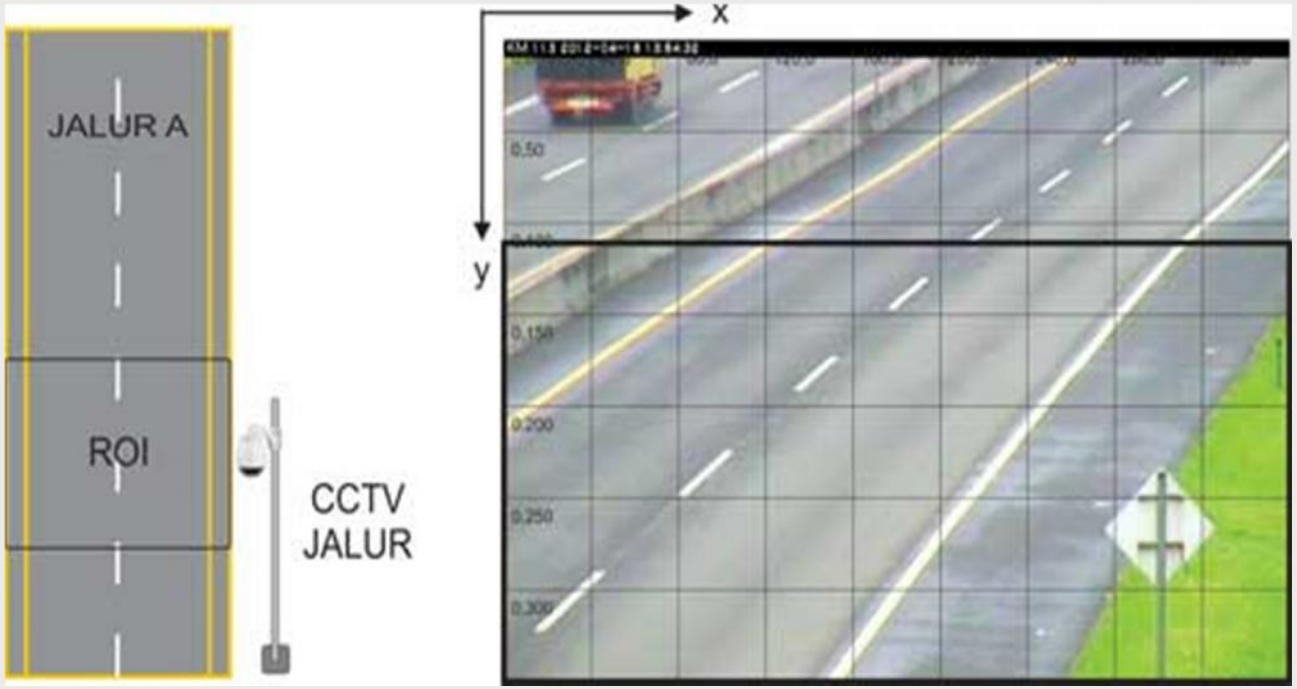



**Penentuan Titik Untuk Pembuatan ROI**

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 2. PENENTUAN *REGION OF INTEREST*

## 2.3 Membuat gambar persegi panjang




***Region of Interest***

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 2. PENENTUAN *REGION OF INTEREST*

## Algoritma untuk menentukan ROI

```
Kamus
   ImageHeight : integer
   ImageWidth : integer
   sample : integer
   roiStartX, roiStartY, roiEndX, roiEndX : integer
   object : Rectangle
   sampleObjects : array[] of Rectangle

Struct Rectangle
   Height, Width : integer
   X, Y : integer (*upper-left coordinate*)

Algoritma
   roiStartX ← ImageWidth
   roiStartY ← ImageHeight
   roiEndX ← 0
   roiEndY ← 0
 Repeat sample
      object ← sampleObjects[sample]

      If (object.X < roiStartX)
         roiStartX ← object.X (*top-left x-coordinate *)
      EndIf
```

```
If (object.Y < roiStartY)
         roiStartY ← object.Y (*top-left y-coordinate *)
      EndIf

      If ((object.X + object.width) > roiEndX)
         roiEndX ← object.X + object.width  (*bottom-right
x-coord *)
      EndIf

      If ((object.Y + object.height) > roiEndY)
         roiEndY ← object.Y + object.height (*bottom-right
y-coord *)
      EndIf

      sample ← sample + 1
   Until (sample = 15)

   ShowRoi(roiStartX,roiStartY,roiEndX,roiEndX)
End.
```

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 3. *GRAYSCALING*



**Citra Berwarna RGB**



**Citra Grayscale**

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 3. *GRAYSCALING*

Formulasi grayscale untuk satu piksel yang mengandung tiga warna RGB adalah:

***Grayscale* = 0.2125*red + 0.7154*green + 0.0721*blue.**

```
Kamus
  ImageHeight : integer
  ImageWidth : integer
  Image : array[ImageHeight][ ImageWidth] of integer
  RGBImage : array[ImageHeight][ ImageWidth] of integer
  GrayImage : array[ImageHeight][ImageWidth] of integer

Algoritma
  For i ← 0 to ImageWidth do
    For j ← 0 to ImageHeight do
      RGBImage[i][j] ← Image[i][j]
      GrayImage[i][j] ←
        0.2125 * RGBImage[i][j].R +
        0.7154 * RGBImage[i][j].G +
        0.0721 * RGBImage[i][j].B
    Endfor
  Endfor
End.
```

**Algoritma Proses *Grayscale***

UNIVERSITAS KOMPUTER INDONESIA UNIKOM
JASAMARGA Indonesia Highway Corp.

# 4. Perhitungan NSSD

**Langkah-langkah Metode NSSD adalah sebagai berikut:**

1. Citra *background*
   Tentukan citra *background* dan nilai pikselnya. Citra ini harus sudah dalam bentuk citra abu-abu.
2. Tentukan citra frame dan nilai pikselnya
   Tentukan citra *frame* dan nilai pikselnya. Citra ini juga harus sudah dalam bentuk citra abu-abu.
3. Untuk setiap frame, lakukan pemrosesan setiap piksel yang ada di dalam ROI menggunakan rumus NSSD berikut:

$$NSSD = \frac{\sum_{i \epsilon Q} \left( I_C(i) - I_B(i) \right)^2}{N}$$

Keterangan :
$I_C(i)$ = intensitas frame pada pixel (i)
$I_B(i)$ = intensitas background pada pixel (i)
Q = detection region/ROI (Region of Interest)
N = jumlah pixel pada area Q

4. Tentukan nilai rata-rata NSSD.

ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 4. Perhitungan NSSD

```
Procedure ProsesNSSD(image : Bitmap)
Kamus
  backgroundImage : Bitmap
  frameImage : Bitmap
  bitmapData : BitmapData
  height, width : integer

procedure OverlayImage()
procedure DifferenceFilter()
procedure ThresholdFilter()

Algoritma
  (* Untuk frame pertama dianggap belum ada objek *)
  If (backgroundImage = NULL)
    backgroundImage ← image
    width ← image.Width
    height ← image.Height
  EndIf

  frameImage ← image
  OverlayImage(backgroundImage,frameImage)
  bitmapData ← DifferenceFilter(backgroundImag,frameImage)
  ThresholdFilter(bitmapData)
End.
```

**ALGORITMA**

# 5. *Thresholding*

*Thresholding* atau penentuan nilai ambang batas untuk memastikan apakah runtunan citra frame merupakan objek yang bergerak atau bukan

1. Nilai kuadrat selisih (*square difference*) > 100
   Piksel dalam citra frame yang memiliki nilai kuadrat selisih yang lebih besar dari 100 dapat dipastikan merupakan piksel objek. Sebaliknya nilai yang lebih kecil dari 100 merupakan bagian dari *background* atau bias.
2. Nilai NSSD > 800
   Kumpulan piksel dalam frame yang memiliki nilai rata-rata NSSD lebih 800 merupakan sebuah objek yang bergerak.

# 6. Identifikasi Bentuk Objek

Contoh identifikasi bentuk objek



# 6. Identifikasi Bentuk Objek

Proses penggambaran bentuk persegi sebuah objek bergerak adalah sebagai berikut:

1. Identifikasi bentuk persegi objek
   Setiap citra *grayscale* dalam frame memiliki piksel-piksel dengan koordinat (x,y) masing-masing. Identifikasi bentuk persegi dilakukan dengan mencari titik terluar dari setiap objek yang terdapat di dalam citra frame.
2. Penggambaran objek sebagai persegi
   Setelah bentuk persegi didapat maka proses selanjutnya yaitu proses penggambaran persegi pada obyek yang terdeteksi.

# 6. Identifikasi Bentuk Objek

**Algoritma Identifikasi Bentuk Objek**

```
Procedure IdentifikasiObjek()
Kamus
   backgroundImage : Bitmap
   frameImage : Bitmap
   bitmapData : BitmapData
   blobCounter : BlobCounter
   objectsCandidate : array[] of Rectangle

procedure GetObjectRectangles()
Struct Rectangle
   Height, Width : integer
   X, Y : integer (*upper-left coordinate*)
Algoritma
   blobCounter(bitmapData)
   objectsCandidate ←
GetObjectRectangles(bitmapData)
   frameImage ← Release(bitmapData)
   Free(frameImage)

End.
```

ANALISIS PROSES PENGHITUNGAN KENDARAAN

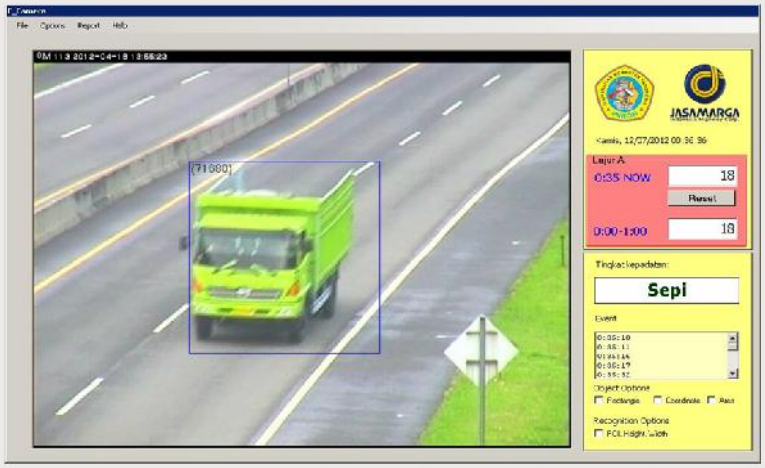
# 7. Penghitungan Kendaraan dan Tingkat Kepadatan

Proses penghitungan kendaraan melalui beberapa proses yang diuraikan sebagai berikut:

1. Penentuan luas objek




Luas minimum adalah 17.000 piksel




Luas maksimum adalah 75.000 piksel.

2. Penambahan Jumlah Kendaraan
   Jika objek terdepan melewati atau meninggalkan ROI, maka jumlah kendaraan dinaikkan satu.
3. Penentuan Tingkat Kepadatan Kendaraan

UNIVERSITAS KOMPUTER INDONESIA UNIKOM

JASAMARGA Indonesia Highway Corp.



ANALISIS PROSES PENGHITUNGAN KENDARAAN

# 7. Penghitungan Kendaraan dan Tingkat Kepadatan

## Algoritma penghitungan kendaraan

```
Procedure HitungKendaraan(Bitmap image)

Kamus
   imageHeight, imageWidth : integer
   candidate, movingObject : Rectangle
   bitmapData : BitmapData
   blobCounter : BlobCounter
   objectsCandidate : array[] of Rectangle
   idx, counter, trafficDensity : integer
   objectGoingToWest, isLeaveRoi, isCounted : boolean

Struct Rectangle
   Height, Width : integer
   X, Y : integer (*upper-left coordinate*)

Algoritma
   counter ← 0
   objectsCandidate ← GetObjectRectangles(bitmapData)
   If (objectsCandidate.Length <> 0)
      Repeat idx
         candidate = objectsCandidate[idx]
         If (isKendaraan(candidate.Height *
candidate.Width)
            If (objectGoingToWest = TRUE)
               If (candidate.Y > roiStartY)
                  If (candidate.X < movingObject.X)
                     movingObject = candidate
                  EndIf
               EndIf
            EndIf
         EndIf
      Until (idx = objectsCandidate.Length)
 If (objectGoingToWest = TRUE)
         If (movingObject.X = roiStartX)
            isLeaveRoi ← TRUE
            isCounted ← TRUE
            movingObject ← NULL
```

```
ElseIf (movingObject.X > roiStartX)
            isCounted ← FALSE
         EndIf
      EndIf
      If (isLeaveRoi = TRUE And isCounted = FALSE)
         counter ← counter + 1
         isLeaveRoi ← TRUE
         isCounted ← FALSE
         addLalin()
      EndIf
   EndIf
End.
Function isKendaraan(pixel: integer) : boolean

Algoritma
   if (pixel > 17000 And pixel < 75000)
      return true
   else
      return false
End.

Procedure addLalin()

Kamus
   Query: array[] of character
   Lalin l Lalin()
   sqlConnection MySqlConnection()

Algoritma
   Query = "INSERT INTO TblLalin(IdCabang, IdPetugas,
IdCamera, Tanggal, Panjang, Lebar, Luas, Status) VALUES (
l.IdCabang, l.IdPetugas, l.IdCamera,
'l.EventDate.ToString("yyyy-MM-dd HH:mm:ss")',
l.Vhcl.Height, l.Vhcl.Width,(l.Vhcl.Height * l.Vhcl.Width),
l.Status)
   MySqlCommand(Query, sqlConnection)
End.
```





# 7. Penghitungan Kendaraan dan Tingkat Kepadatan

### Algoritma tingkat kepadatan

```
Function VehicleDensity (numOfVehiclePerTimeUnit : integer) : array[] of
character

Kamus
   vhclDensity : array[] of character

Algoritma

   If (numOfVehiclePerTimeUnit <= 1400)
         vhclDensity= “Sepi”
   ElseIf (numOfVehiclePerTimeUnit > 1400 And numOfVehiclePerTimeUnit <= 2275)
         vhclDensity = “Sedang”
   ElseIf (numOfVehiclePerTimeUnit >= 2275 And numOfVehiclePerTimeUnit < 2975)
         vhclDensity = “Padat”
   Else
         vhclDensity = “Macet”
   EndIf
   Return vhclDensity
End.
```

## ANALISIS KEBUTUHAN NON FUNGSIONAL

# 1. Analisis Pengguna

### Tabel Pengguna Sistem yang sedang berjalan

| Pengguna | Tugas | Keterangan |
|---|---|---|
| Petugas Sentra Komunikasi | - Memonitor dan mengendalikan kamera CCTV.<br>- Menerima dan meneruskan laporan dari lapangan.<br>- Memprediksi kondisi lalu lintas.<br>- Membuat laporan yang ditulis manual pada form yang telah disediakan. | Petugas pemantau lalu lintas yang berada di cabang Jasa Marga (Purbaleunyi) |
| Unit Sarana Elektronik | Memonitor kondisi perangkat pemantauan agar tetap terjaga dan berfungsi dengan baik. | Unit pemeliharaan dan perbaikan perangkat pemantauan. |
| JMTIC (Jasa Marga Trafic Information System) | Memonitor kamera CCTV | Pusat Informasi yang berada di Kantor Pusat Jasa Marga. |
| Kepolisian | Memonitor kamera CCTV | Instansi yang bekerja sama dengan Jasa Marga dalam pemantauan lalu lintas di jalan tol |
| Kasubag YANLALIN | Memeriksa laporan kepadatan lalu lintas dari petugas sentra komunikasi | Atasan dari petugas sentra komnunikasi |

### Tabel Pengguna Sistem Baru

| Pengguna | Tugas | Keterangan |
|---|---|---|
| Petugas Sentra Komunikasi | - Memonitor dan mengendalikan kamera CCTV.<br>- Menerima dan meneruskan laporan dari lapangan.<br>- Membuat laporan kepadatan lalu lintas dari aplikasi pemantauan kepadatan lalu lintas. | Petugas dalam melakukan pemantauan dan pembuatan laporan dibantu oleh aplikasi pemantauan lalu lintas. |

ANALISIS KEBUTUHAN NON FUNGSIONAL

# 2. Analisis Kebutuhan Perangkat Keras

**Tabel Spesifikasi perangkat keras sistem yang sedang berjalan**

| Infrastruktur | Nama Perangkat Keras | Deskripsi |
|---|---|---|
| Jalur | Kamera CCTV | Kamera AXIS |
| Server | CPU | IBM type X3650 |
| | Monitor | LCD Monitor 17” |
| | Harddisk | 10 TB |
| Client | CPU | P4 1.8 Ghz |
| | RAM | 512 MB |
| | Monitor | LCD Monitor 42” |
| | Harddisk | 20 GB |
| Perangkat keras pendukung komunikasi data | Kabel FO | |
| | OTB Box | |
| | Converter FO | |
| | Kabel UTP/STP | |
| | Switch HUB | |
| | Switch HUB FO | |

**Tabel Spesifikasi perangkat keras sistem yang akan dibangun**

| Infrastruktur | Nama Perangkat Keras | Deskripsi |
|---|---|---|
| Jalur | Kamera CCTV | Kamera AXIS |
| Server | CPU | IBM type X3650 |
| | Monitor | LCD Monitor 17” |
| | Harddisk | 10 TB |
| Client | CPU | P4 1.8 Ghz |
| | RAM | 512 MB |
| | Monitor | LCD Monitor 42” |
| | Harddisk | 20 GB |
| Perangkat keras pendukung komunikasi data | Kabel FO | |
| | OTB Box | |
| | Converter FO | |
| | Kabel UTP/STP | |
| | Switch HUB | |
| | Switch HUB FO | |

UNIVERSITAS KOMPUTER INDONESIA UNIKOM

JASAMARGA Indonesia Highway Corp.

ANALISIS KEBUTUHAN NON FUNGSIONAL

# 3. Analisis Kebutuhan Perangkat Lunak

**Tabel Spesifikasi Perangkat lunak pada sistem yang Berjalan Saat Ini**

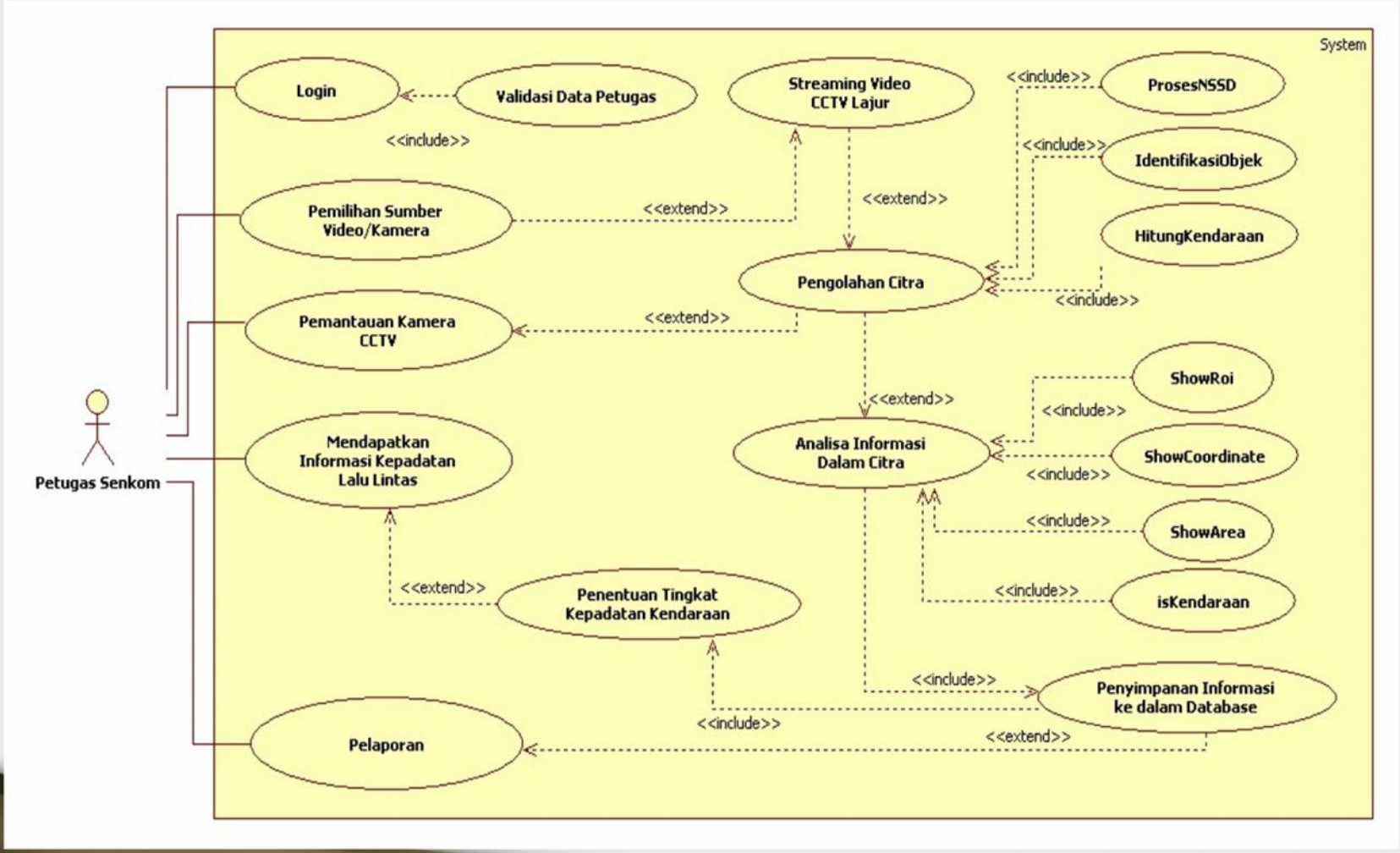
| Infrastruktur | Nama Perangkat Lunak | Deskripsi |
|---|---|---|
| Server | Sistem Operasi | Windows Server 2008 R2 |
| | AXIS OCS Server | Aplikasi Recording video kamera CCTV |
| Client | Sistem operasi | Windows 7 |
| | AXIS OCS Client | Aplikasi untuk menampilkan video kamera CCTV |
| Pendukung komunikasi data | | Jaringan LAN (Local Area Network) |

**Tabel Spesifikasi Perangkat lunak pada sistem yang akan dibangun**

| Infrastruktur | Nama Perangkat Lunak | Deskripsi |
|---|---|---|
| Server | Sistem Operasi | Windows Server 2008 R2 |
| | AXIS OCS Server | Aplikasi Recording video kamera CCTV |
| | IDE | Visual Studio 2008(C#) |
| | Framework Image Processing | AForge.Net Framework 2.2.4 |
| | DBMS | MySql 5.5.8 |
| | SQL Connector | MySQL Connector Net 6.3.8 |
| Client | Sistem operasi | Windows 7 |
| | AXIS OCS Client | Aplikasi untuk menampilkan video kamera CCTV |
| | Framework | Framework .Net 3.5 |
| Pendukung komunikasi data | | Jaringan LAN (Local Area Network) |

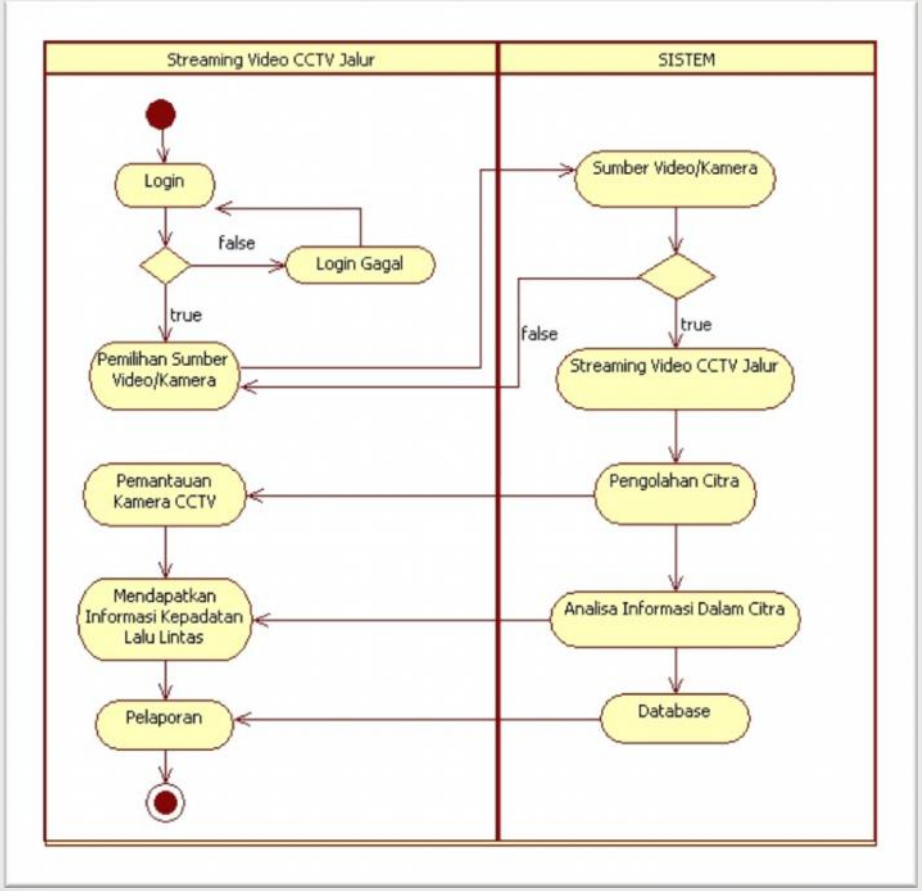
## ANALISIS KEBUTUHAN FUNGSIONAL

# 1. *Use Case Diagram*

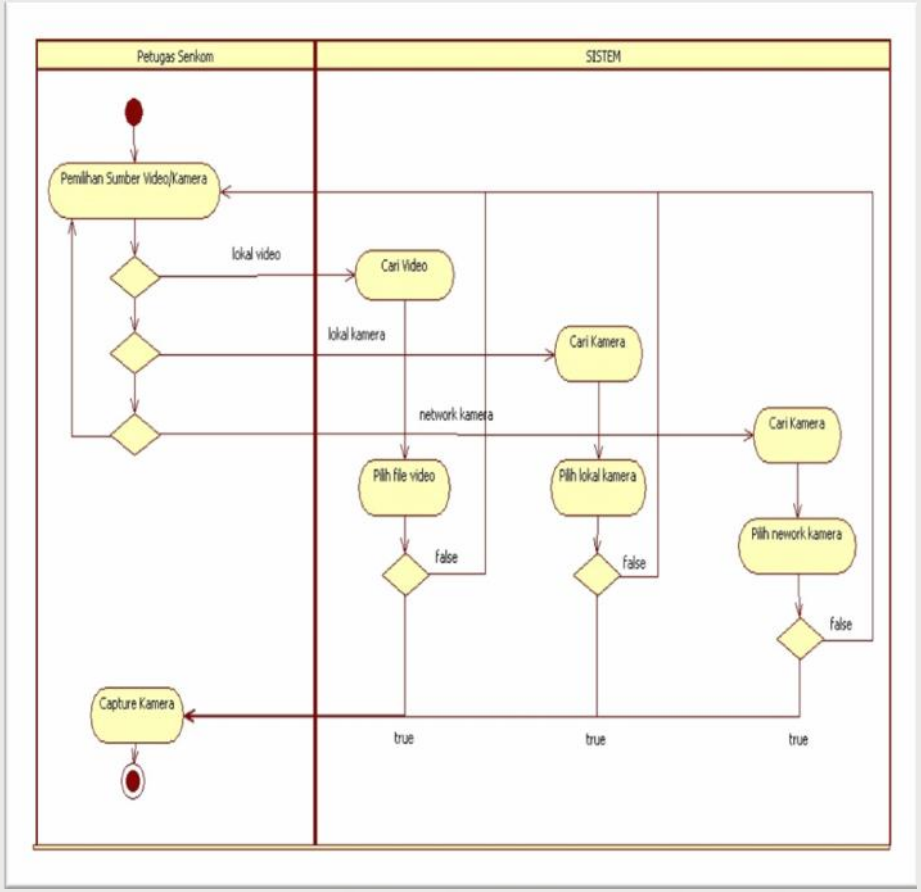


ANALISIS KEBUTUHAN FUNGSIONAL

# 2. *Activity Diagram*



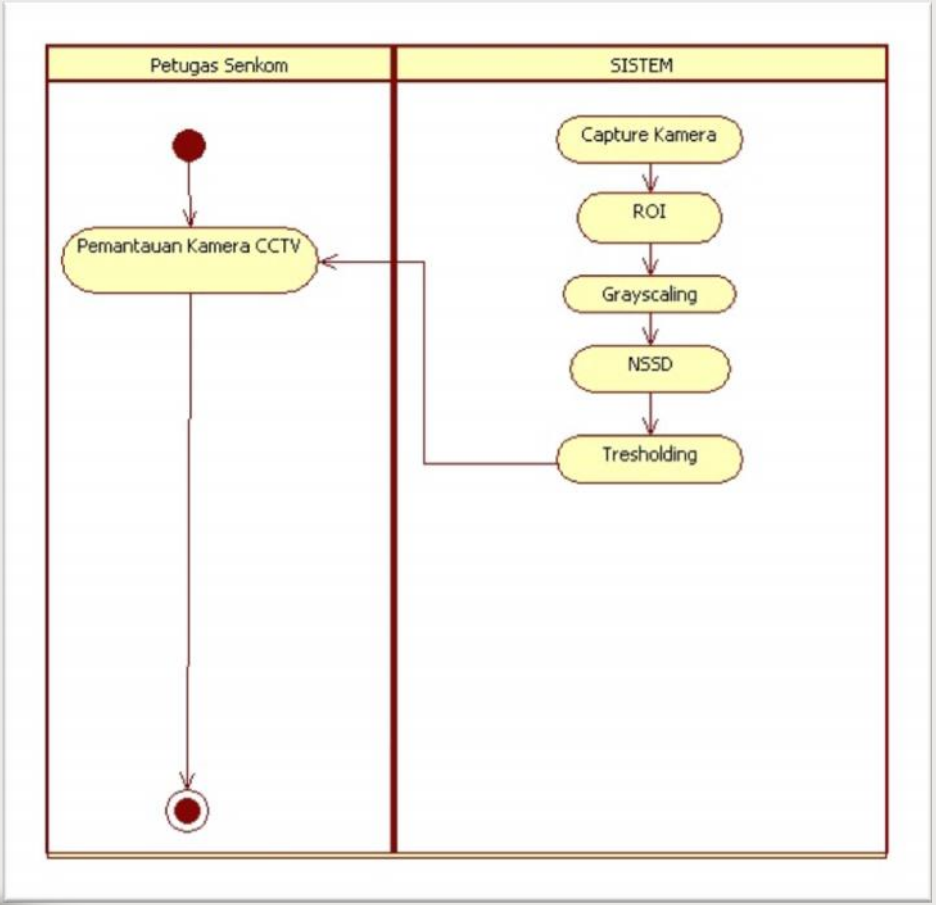

***Activity diagram* Sistem Pemantauan Lalu Lintas Otomatis**



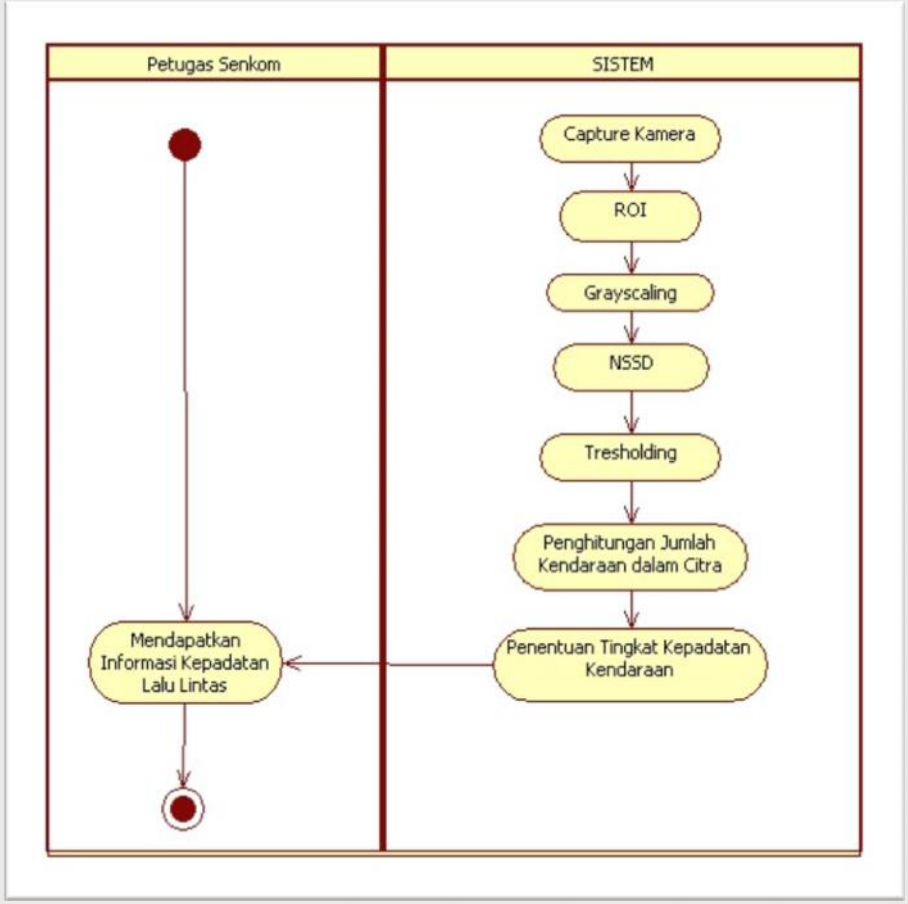

***Activity diagram* Pemilihan Sumber Video/Kamera**

ANALISIS KEBUTUHAN FUNGSIONAL

# 2. *Activity Diagram*



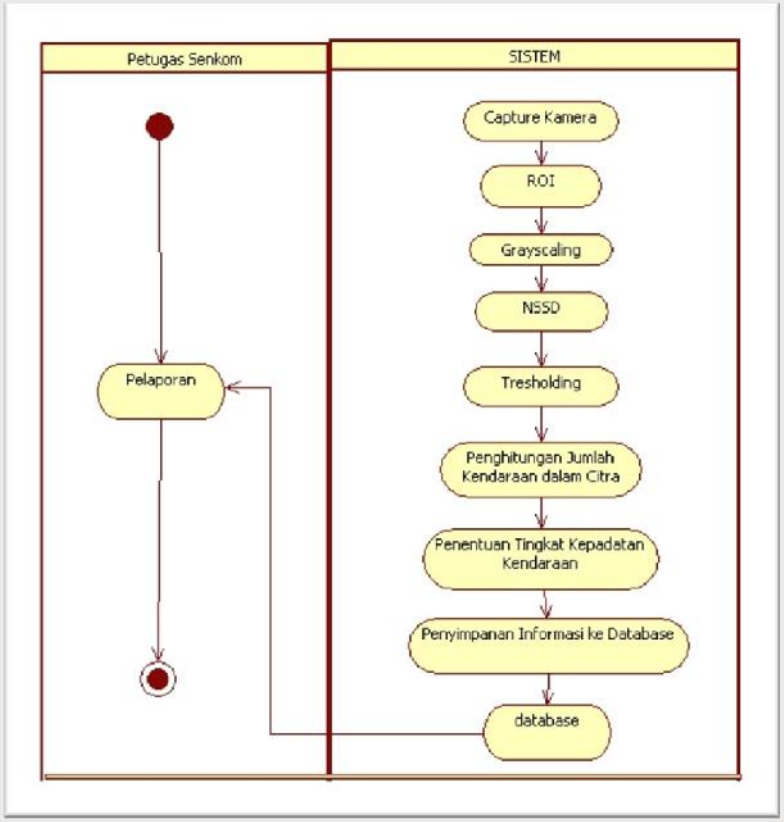

***Activity diagram* Pemantauan Kamera CCTV**




***Activity diagram* Mendapatkan Informasi Kepadatan Lalu Lintas**

ANALISIS KEBUTUHAN FUNGSIONAL

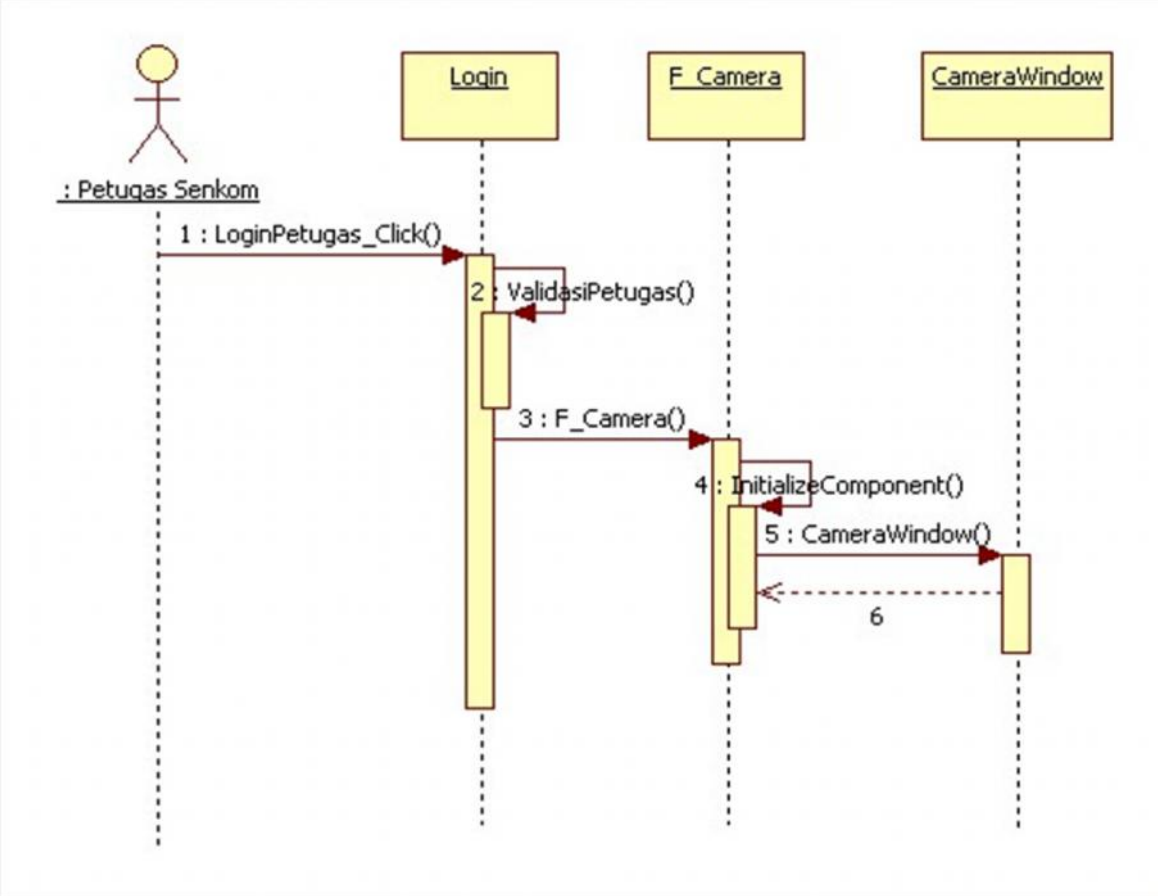
# 2. *Activity Diagram*




***Activity diagram* Pelaporan**

ANALISIS KEBUTUHAN FUNGSIONAL

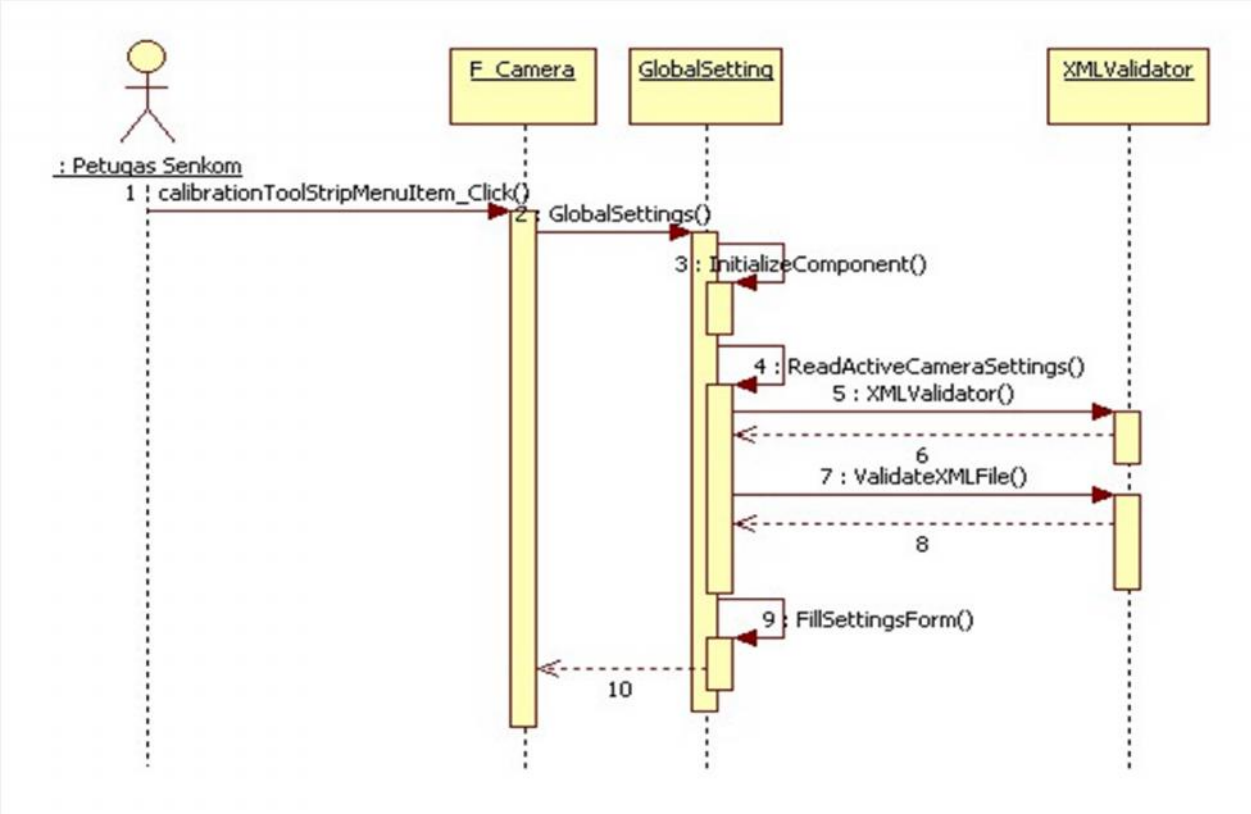
# 3. *Sequence Diagram*




***Sequence Diagram Login***

ANALISIS KEBUTUHAN FUNGSIONAL

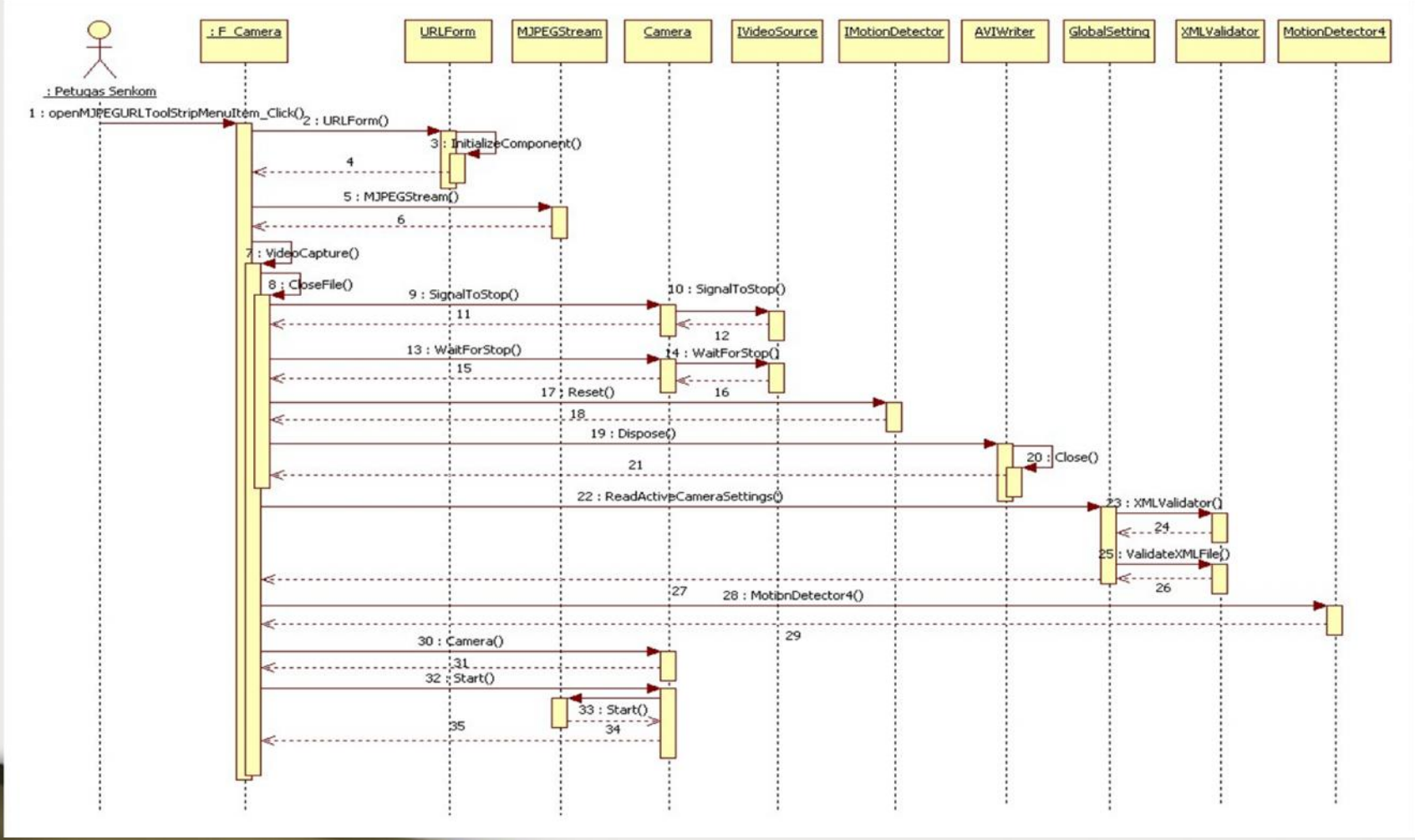
# 3. *Sequence Diagram*




***Sequence Diagram Show Options***

ANALISIS KEBUTUHAN FUNGSIONAL

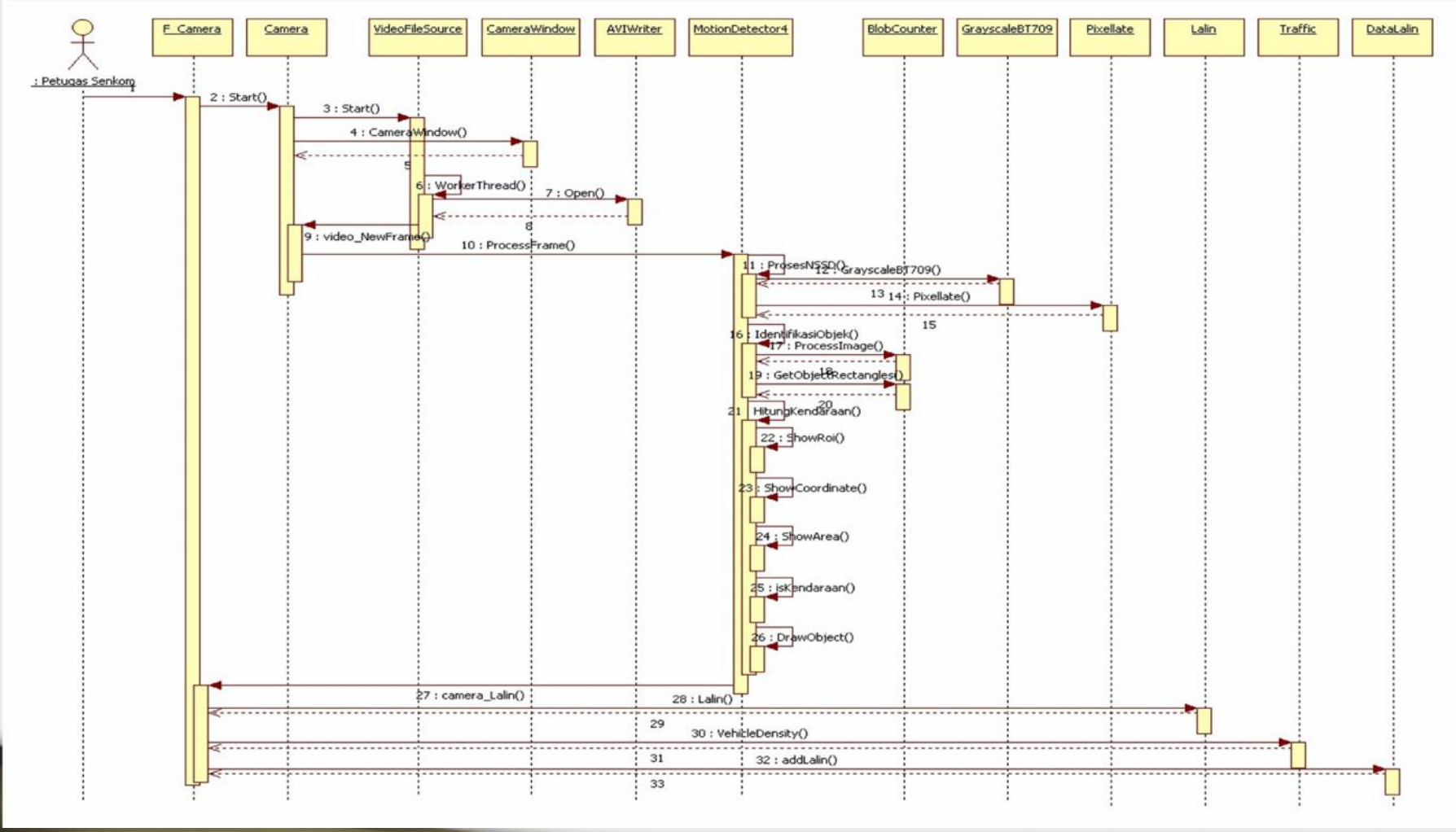
# 3. *Sequence Diagram*




***Sequence Diagram Streaming Video CCTV***

ANALISIS KEBUTUHAN FUNGSIONAL

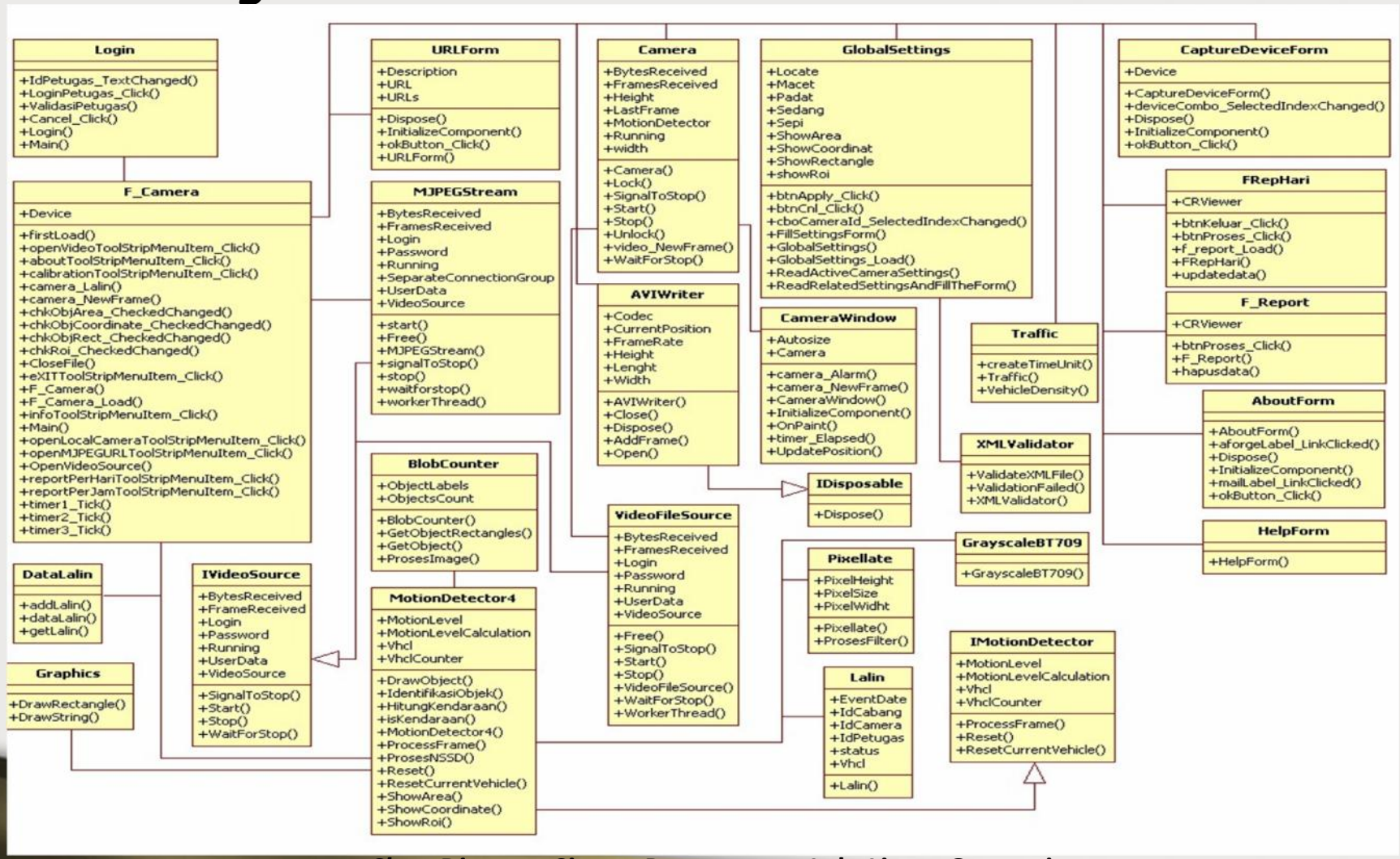
# 3. *Sequence Diagram*




***Sequence Diagram* Pengolahan Citra dan Analisa Citra**

ANALISIS KEBUTUHAN FUNGSIONAL

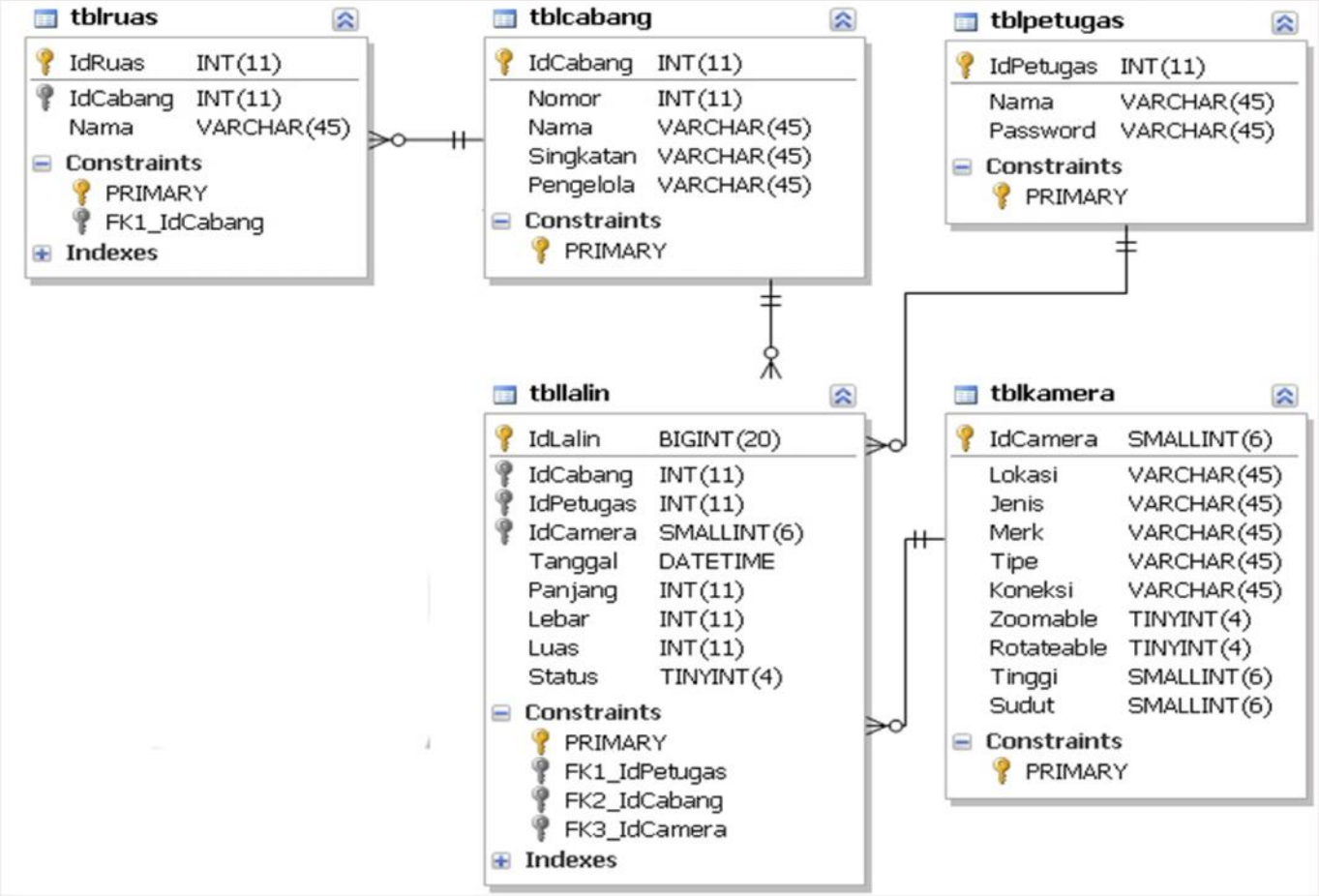
# 4. *Class Diagram*




***Class Diagram* Sistem Pemantauan Lalu Lintas Otomatis**

PERANCANGAN DATABASE

# 1. Diagram Relasi



PERANCANGAN DATABASE

# 2. Struktur Tabel *Database*

**Struktur Tabel Ruas**

| Field | Type | Keterangan |
|---|---|---|
| IdRuas | INT (11) | PRIMARY KEY |
| IdCabang | INT (11) | FOREIGN KEY, Ref tblcabang(idCabang) |
| Nama | VARCHAR (45) | |

**Struktur Tabel Cabang**

| Field | Type | Keterangan |
|---|---|---|
| IdCabang | INT (11) | PRIMARY KEY |
| Nama | VARCHAR (45) | |
| Singkatan | VARCHAR (45) | |
| Pengelola | VARCHAR (45) | |

**Struktur Tabel Kamera**

| Field | Type | Keterangan |
|---|---|---|
| IdCamera | SMALLINT (6) | PRIMARY KEY |
| Lokasi | VARCHAR (45) | |
| Jenis | VARCHAR (45) | |
| Merk | VARCHAR (45) | |
| Tipe | VARCHAR (45) | |
| Koneksi | VARCHAR (45) | |
| Zoomable | TINYINT (4) | |
| Rotateable | TINYINT (4) | |
| Tinggi | SMALLINT (6) | |
| Sudut | SMALLINT (6) | |

**Struktur Tabel Petugas**

| Field | Type | Keterangan |
|---|---|---|
| IdPetugas | INT (11) | PRIMARY KEY |
| Nama | VARCHAR (45) | |
| Password | VARCHAR (45) | |

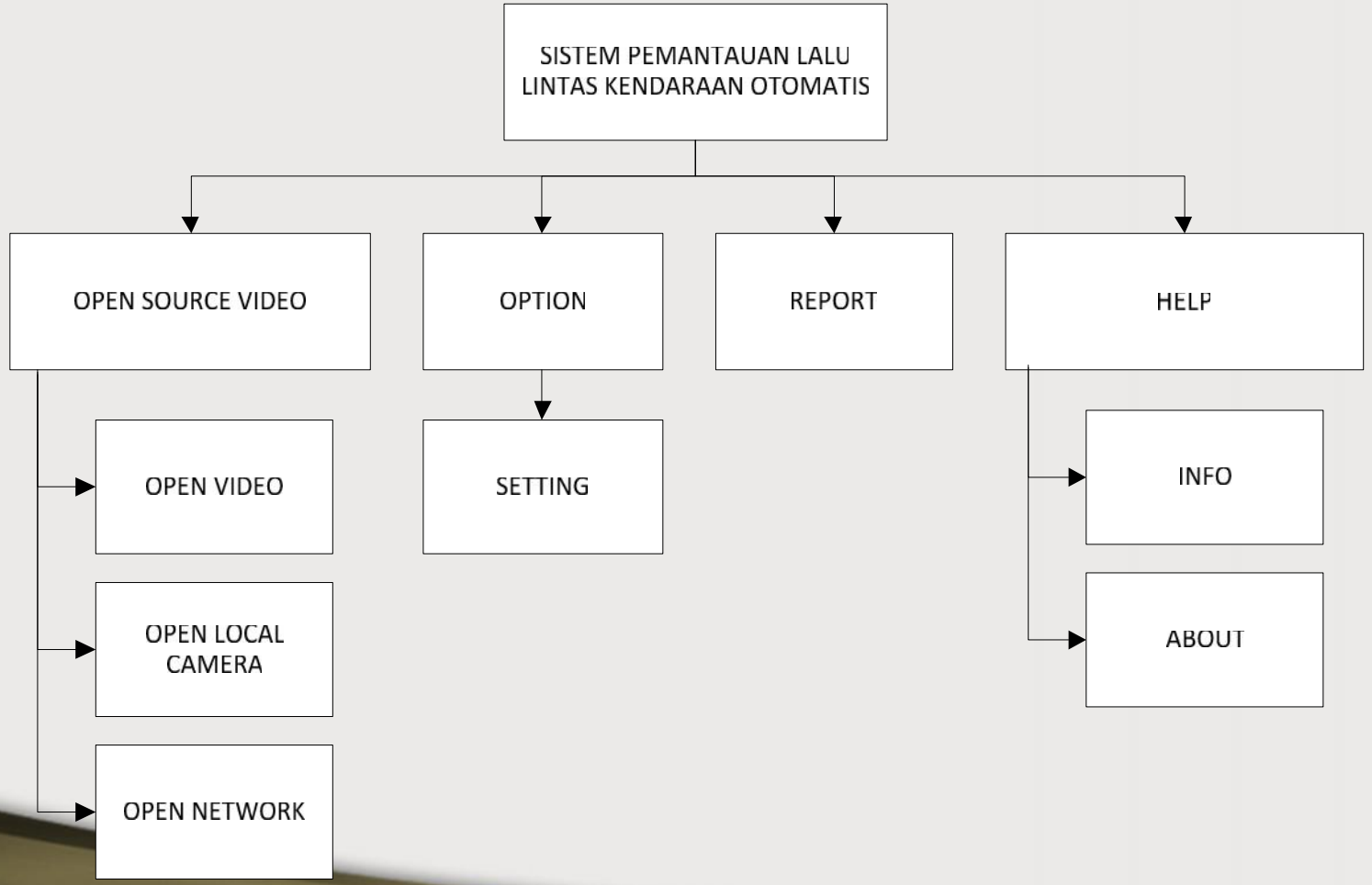
**Struktur Tabel Lalin**

| Field | Type | Keterangan |
|---|---|---|
| IdLalin | BIGINT (20) | PRIMARY KEY |
| Tanggal | DATETIME | |
| IdCabang | INT (11) | FOREIGN KEY, Ref tblcabang(idCabang) |
| IdCamera | SMALLINT (6) | FOREIGN KEY, Ref tblkamera(IdCamera) |
| IdPetugas | INT (11) | FOREIGN KEY, Ref tblpetugas(IdPetugas) |
| Panjang | INT (11) | |
| Lebar | INT (11) | |
| Luas | INT (11) | |
| Status | TINYINT (4) | |

PERANCANGAN ANTARMUKA

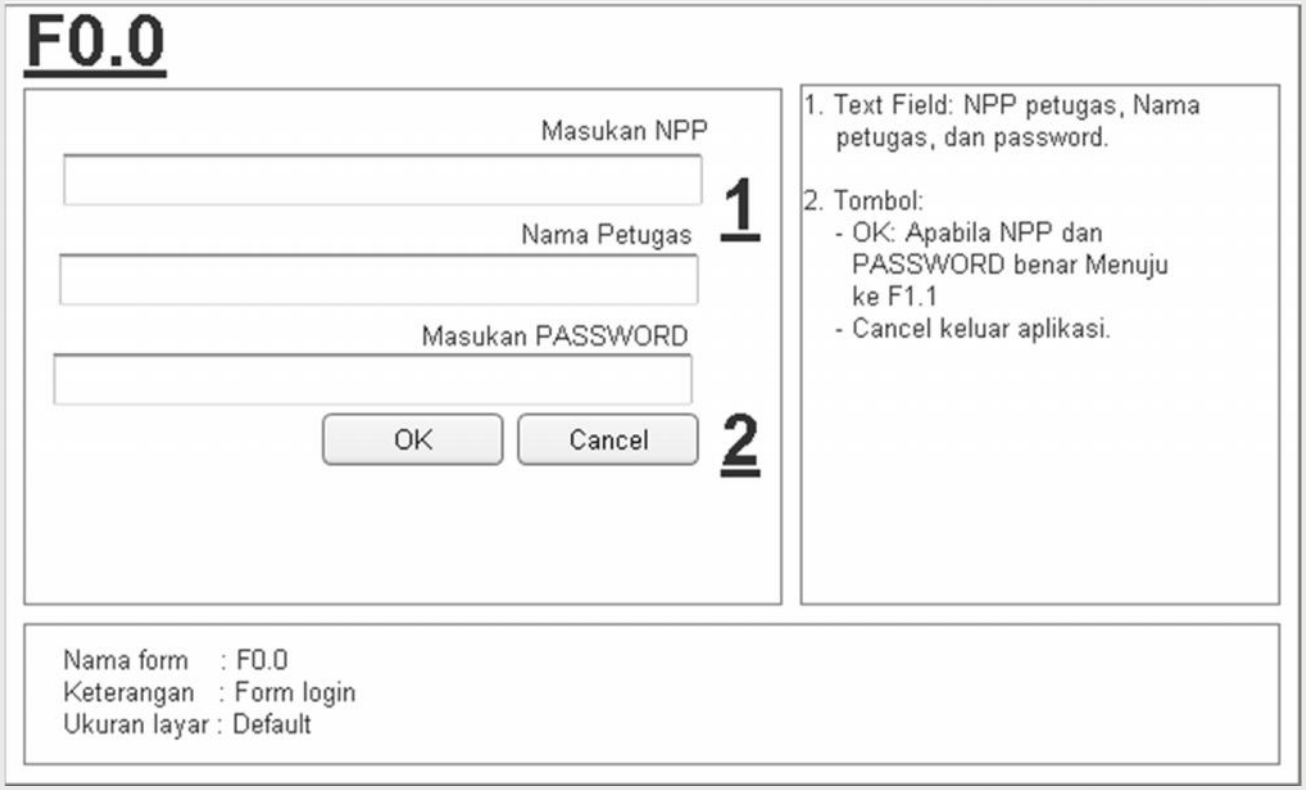
# 1. Perancangan Menu

**Struktur Menu Aplikasi Pemantauan Lalu Lintas Kendaraan Otomatis**



PERANCANGAN ANTARMUKA

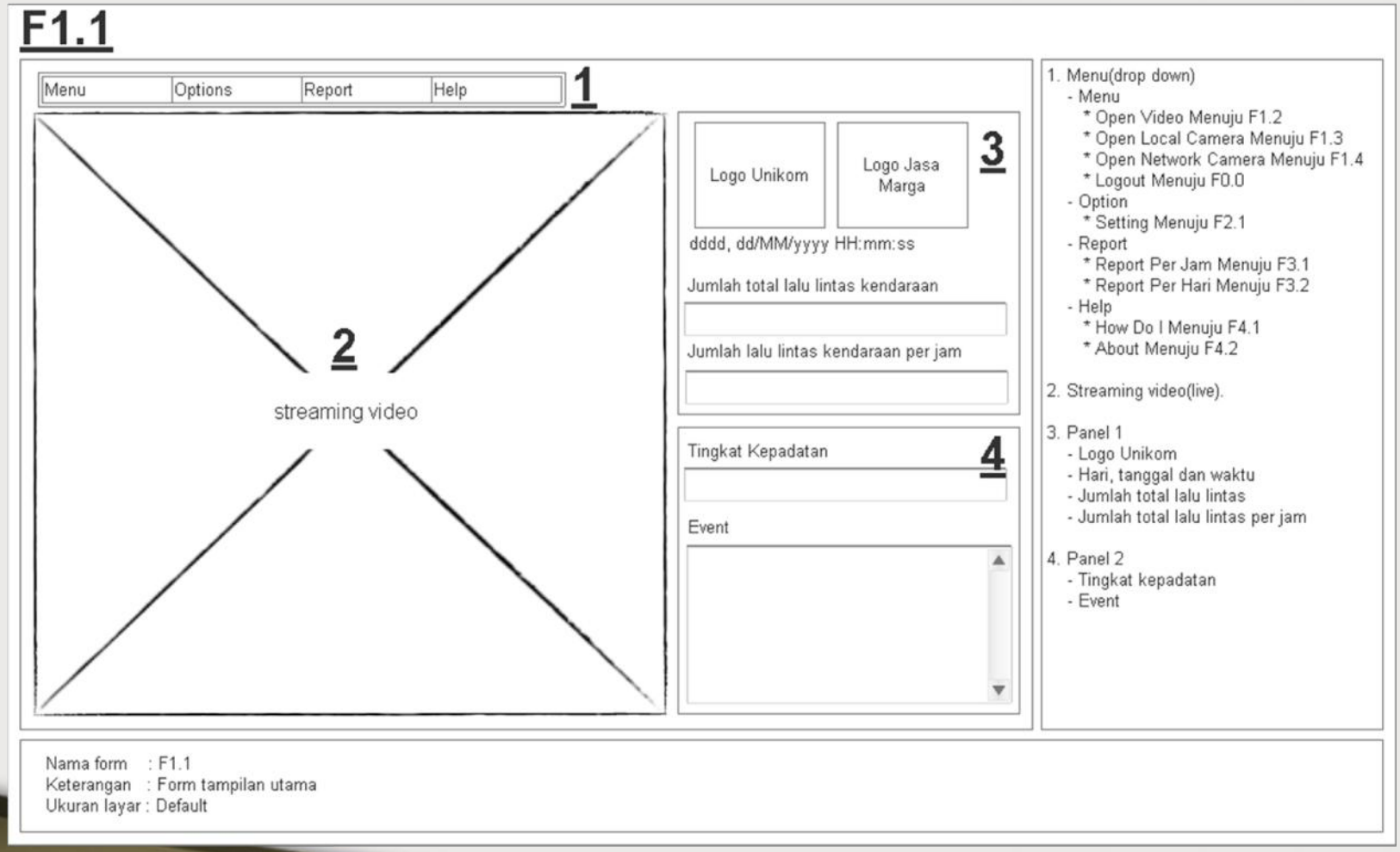
# 2. Perancangan Form




**Form Login(F0.0)**

PERANCANGAN ANTARMUKA

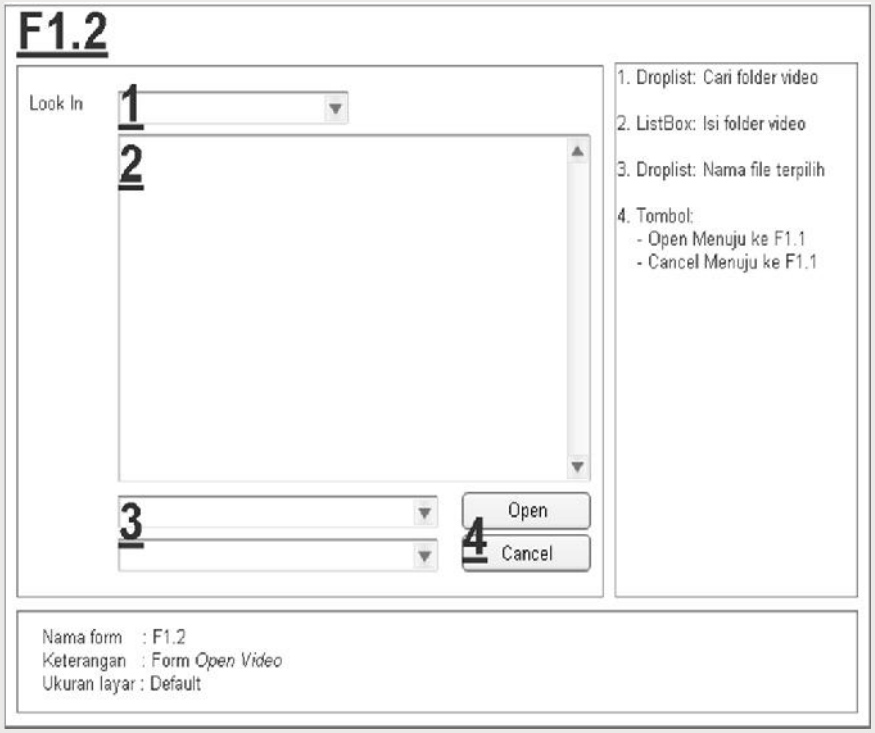
# 2. Perancangan Form



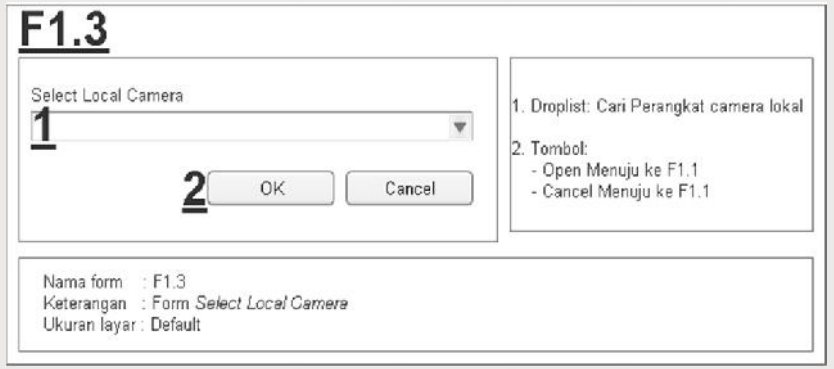

**Form Tampilan Utama(F1.1)**

PERANCANGAN ANTARMUKA

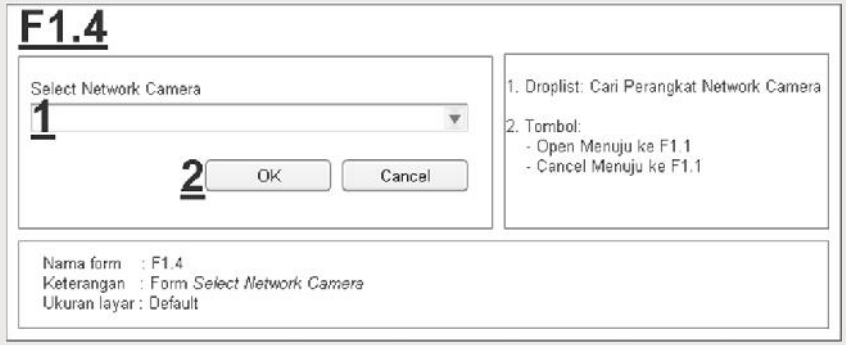
# 2. Perancangan Form




**Form *Open Video*(F1.2)**



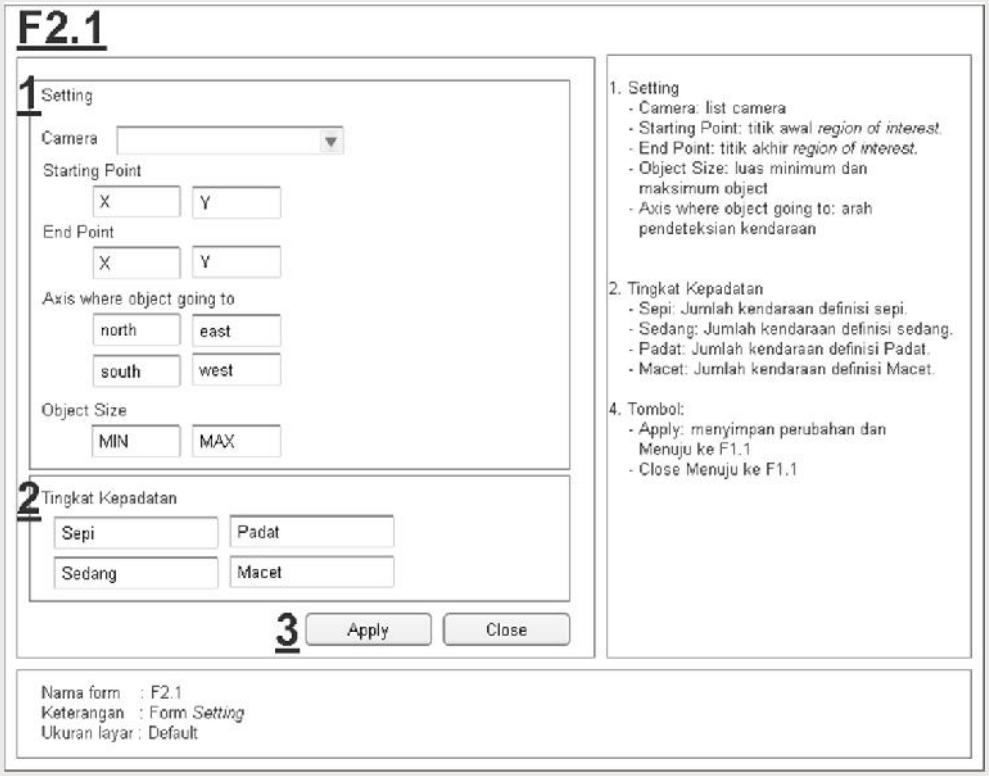

**Form *Local Camera*(F1.3)**



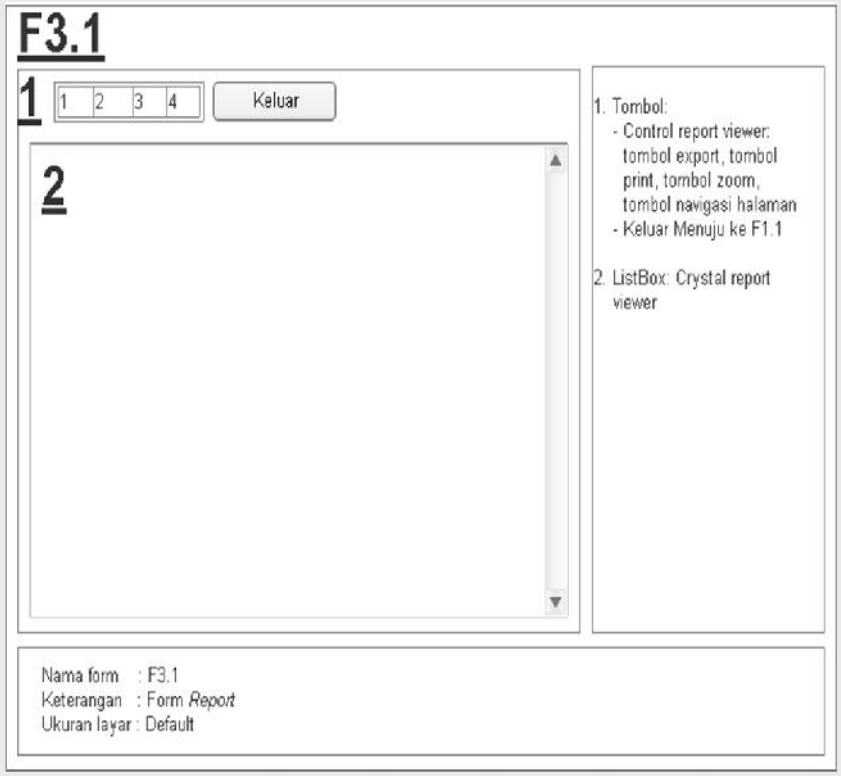

**Form *Network Camera* (F1.4)**

PERANCANGAN ANTARMUKA

# 2. Perancangan Form



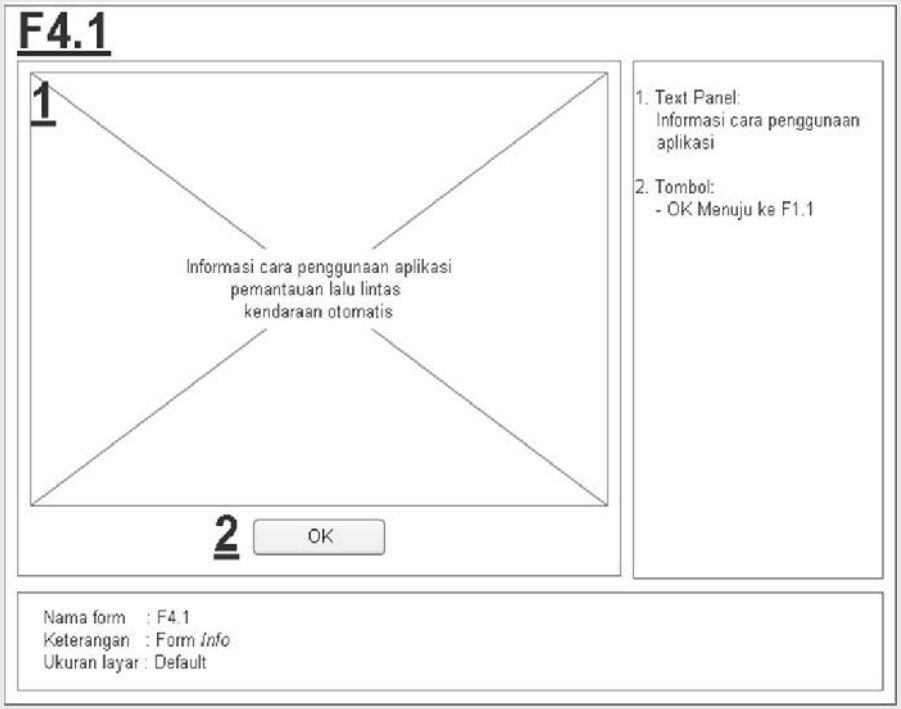

Form *Setting*(F2.1)



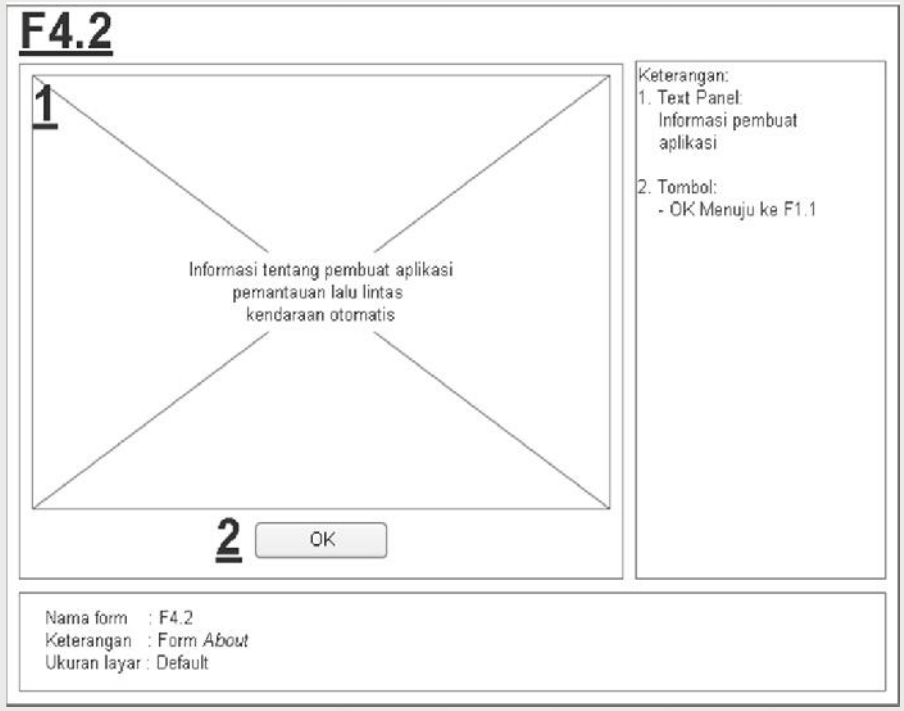

Form *Report*(F3.1)

PERANCANGAN ANTARMUKA

# 2. Perancangan Form



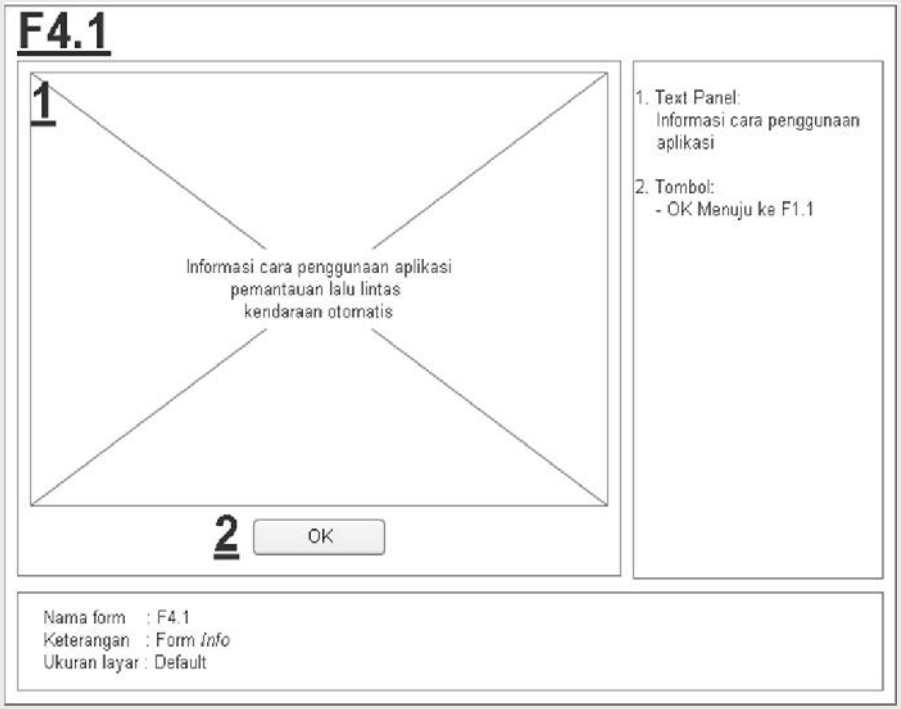

Form *help*(F4.1)



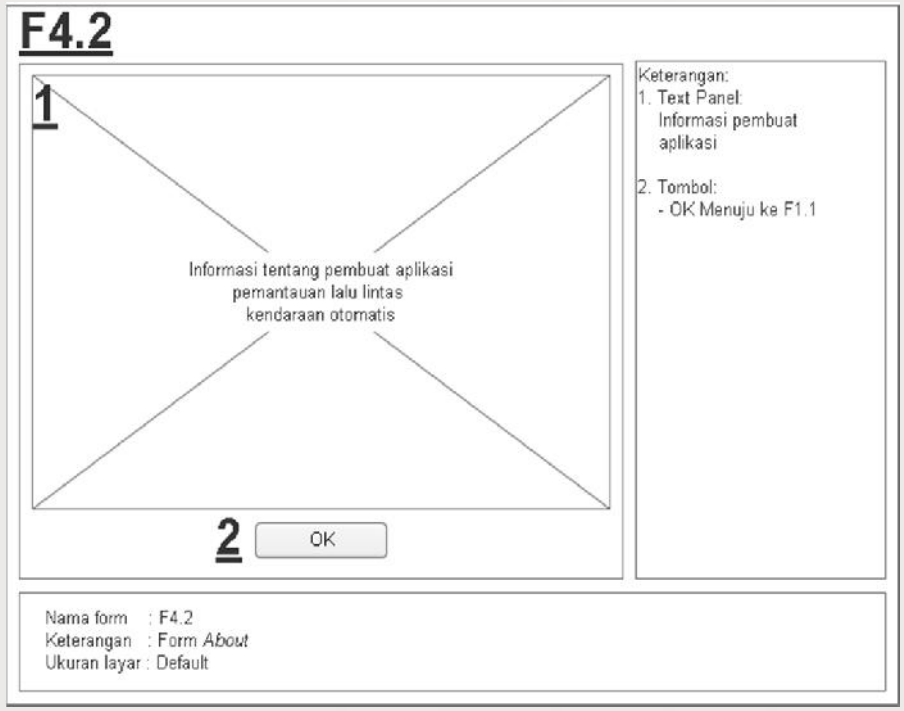

Form *About*(F4.2)

PERANCANGAN ANTARMUKA

# 2. Perancangan Form




Form *help*(F4.1)




Form *About*(F4.2)



4 Implementasi

5 Demo Aplikasi



# TERIMA KASIH