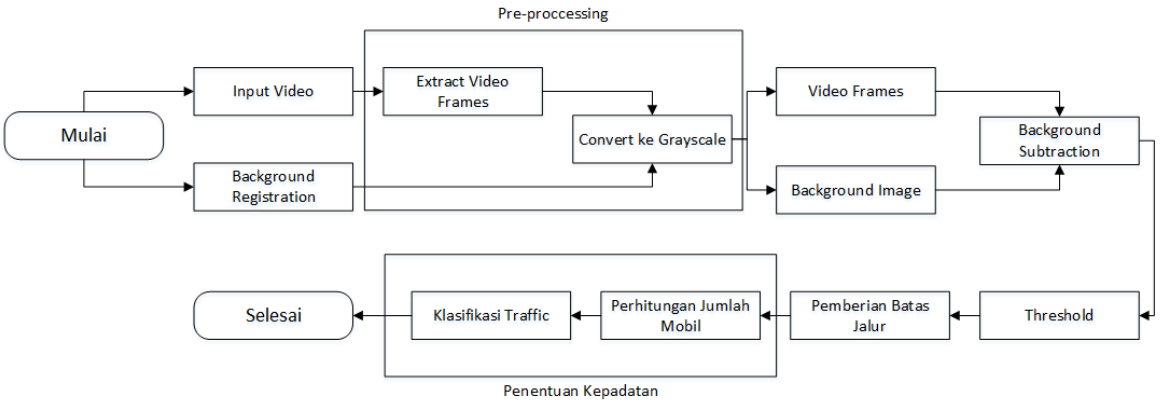
BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Masalah

Analisis masalah pada penelitian ini adalah bagaimana memproses rekaman dari kamera pengawas untuk mengetahui kondisi kepadatan suatu gerbang tol. Penelitian ini membutuhkan perhitungan jumlah mobil yang terekam untuk selanjutnya diproses kembali. Hasil perhitungan jumlah mobil digunakan sebagai data masukan untuk mengetahui kepadatan pada suatu gerbang tol.

3.2 Analisis Proses

Analisis proses yang akan dilakukan pada penelitian ini dibagi menjadi beberapa tahapan. Pertama adalah pre-processing, lalu dilakukan background registration untuk mendapatkan background images. Setelah mendapatkan background images, dilakukan proses background substraction/foreground detection. Hasil dari proses tersebut akan di thereshold. Setelah itu akan dilakukan pemberian garis jalur dan koordinat untuk menghitung jumlah mobil. Jumlah mobil yang terhitung akan kembali dihitung menggunakan teori antrian untuk menentukan keadaan gerbang tol. Hasil dari proses tersebut akan menghasilkan output apakah gerbang tersebut sedang padat atau tidak.



Gambar 3.1 Analisis Proses

3.3 Analisis Data Masukan

Data masukan yang akan dianalisis berupa citra background dan video hasil rekaman kamera pengawas gerbang tol yang didapatkan dari website live streaming PT. Jasa Marga. Pada penelitian ini data masukan yang digunakan adalah video hasil rekaman kamera pengawas pada gerbang tol pasteur. Pada video hasil rekaman tersebut akan dilakukan preprocessing yaitu video dirubah menjadi frame-frame untuk proses konversi ke grayscale setelah itu akan masuk ke tahapan background registration untuk menghasilkan background images setelah itu ke tahapan background subtraction. Setelah itu masuk ke tahapan perhitungan dan klasifikasi traffic.

3.3.1 Background Registration

Background registration merupakan tahap persiapan citra background yang akan digunakan pada proses background subtraction. Nilai citra background ini nantinya akan dikurangkan dengan citra video dimana objek mobil akan dideteksi. Pada penelitian ini digunakan background dari hasil capture rekaman kamera pengawas gerbang tol saat tidak ada kendaraan yang melewat. Terdapat 2 citra background yang digunakan adalah sebagai berikut :



Gambar 3.2 Citra Background 1

Citra background 1 tersebut memliki informasi sebagai berikut :

Tabel 3.1 Informasi Citra Background 1

File Name	Bgtest2.jpg
Format	JPG
File Size	58,7 KB
Dimensions	357 x 271
Horizontal Resolution	96 ppi
Vertical Resolution	96 ppi
Bit Depth	24
Color	RGB

Citra background kedua adalah sebagai berikut :



Gambar 3.3 Citra Background 2

Citra background 2 tersebut memliki informasi sebagai berikut :

Tabel 3.2 Informasi Citra Background 2

File Name	Bgtest2.jpg
Format	JPG
File Size	58,7 KB
Dimensions	357 x 271
Horizontal Resolution	96 ppi
Vertical Resolution	96 ppi
Bit Depth	24
Color	RGB

3.3.2 Preprocessing

Preprocessing merupakan tahap persiapan untuk proses awal. Input yang berupa video diproses per frame.Preprocessing meliputi 2 tahap, yaitu merubah video menjadi frame dan konversi dari RGB ke grayscale.

3.3.2.1 Ekstrak *Video Frames*

Video merupakan gabungan gambar-gambar mati yang dibaca berurutan dalam suatu waktu dengan kecepatan tertentu. Gambar-gambar yang digabung tersebut dinamakan *frame* dan kecepatan pembacaan gambar disebut dengan *frame rate*, dengan satuan fps (*frame per second*). Karena dimainkan dalam kecepatan yang tinggi maka tercipta ilusi gerak yang halus, semakin besar nilai frame rate maka akan semakin halus pergerakan yang ditampilkan. Tahapan awal yaitu mengetahui jumlah frame yang ada pada video.

Sebuah video tentu akan memiliki jumlah *frame* yang beragam tergantung kualitas kamera digunakan. Untuk kamera pengawas biasanya menggunakan jenis kamera standar yang memiliki kualitas hasil rekaman dan jumlah frame dibawah kamera yang biasa digunakan pada smartphone atau kamera profesional.

Pada langkah ekstraksi video frame, dibutuhkan sebuah library yang dapat memanipulasi masukan berupa video. Library yang digunakan adalah *Xuggler*. *Xuggler* merupakan libarary open-source untuk pengembang pemrograman *Java* yang dapat digunakan untuk memanipulasi video hasil rekaman. *Xuggler* dapat melakukan ekstraksi pada sebuah video untuk dijadikan potongan-potongan gambar untuk setiap detiknya.

Pada hasil rekaman kamera pengawas gerbang tol PT.Jasa Marga didapat video dengan informasi sebagai berikut :

Tabel 3.3 Informasi Video

File Name	rec3.avi
Format	Avi
File Size	35,3 MB
Duration	00:01:42
Frame Width	357

Frame Height	271
Data Rate	2589 kbps
Frame Rate	15 frames/second

Untuk mengetahui jumlah frame pada sebuah video dapat dihitung dengan persamaan (7) :

*Jumlah Frame = Frame rate * Duration*

dimana :

Frame rate merupakan jumlah frame tiap detik (frame/second atau f/s)

Duration merupakan durasi video dengan satuan detik (second atau s)

Sehingga berdasarkan informasi pada tabel diatas dapat diketahui bahwa jumlah frame pada video tersebut adalah sebagai berikut :

Duration = 00:01:42 = 0 + 60 + 42 = 102 second

*Jumlah frame = 15 f/s * 102 s = 1530 frames*

Jumlah frame yang tersedia pada video dipengaruhi oleh frame rate dan durasi. Semakin besar frame rate atau durasi dalam menampilkan gambar, maka semakin banyak pula yang diperoleh.

3.3.2.2 Convert Grayscale

Citra background dan citra yang didapat dari frame video hasil rekaman memiliki format gambar RGB. Pada tahap ini setiap frame akan dirubah menjadi grayscale agar lebih mudah diproses pada tahap selanjutnya. Grayscale merupakan warna-warna piksel yang berada dalam rentang gradasi warna hitam dan putih. Untuk merubah gambar RGB menjadi grayscale dapat dilakukan dengan persamaan (8) seperti berikut :

$$\mathrm{Y}(\mathrm{x},\mathrm{y}) = (0{,}229 * \mathrm{R}) + (0{,}587 * \mathrm{G}) + (0{,}114 * \mathrm{B})$$

dimana :

Y = derajat keabuan

R = nilai pixel channel Red

G = nilai pixel channel Green

B = nilai pixel channel Blue

Pada citra background diambil citra ukuran 10x10 pixel pada suatu titik :



Gambar 3.4 Citra 10x10

Nilai pixel citra background 10x10 adalah sebagai berikut :

R:184 G:175 B: 146	R:182 G:171 B:143	R:177 G:166 B:138	R:172 G:161 B:133	R:172 G:161 B:133	R:177 G:165 B:139	R:183 G:171 B:145	R:181 G:175 B:143	R:188 G:182 B:150	R:189 G:183 B:149
R:180 G:1171 B: 142	R:177 G:166 B:138	R:173 G:162 B:134	R:172 G:161 B:133	R:175 G:164 B:136	R:180 G:168 B:142	R:184 G:172 B:146	R:183 G:177 B:145	R:187 G:181 B:149	R:187 G:181 B:149
R:176 G:167 B: 138	R:172 G:161 B:133	R:172 G:161 B:133	R:174 G:163 B:135	R:179 G:168 B:140	R:183 G:171 B:145	R:186 G:179 B:151	R:186 G:179 B:151	R:187 G:180 B:152	R:184 G:177 B:148
R:173 G:164 B: 135	R:170 G:159 B:131	R:172 G:161 B:133	R:178 G:167 B:139	R:185 G:174 B:146	R:189 G:177 B:151	R:188 G:181 B:152	R:188 G:181 B:152	R:185 G:178 B:149	R:181 G:174 B:145
R:169 G:160 B: 131	R:168 G:157 B:129	R:172 G:161 B:133	R:180 G:169 B:141	R:187 G:176 B:148	R:189 G:177 B:151	R:187 G:180 B:152	R:187 G:180 B:152	R:182 G:175 B:147	R:177 G:170 B:142
R:167 G:158 B: 129	R:170 G:159 B:131	R:175 G:164 B:136	R:180 G:169 B:141	R:184 G:172 B:146	R:185 G:173 B:147	R:186 G:179 B:151	R:186 G:179 B:151	R:180 G:173 B:145	R:174 G:167 B:138
R:169 G:160 B: 131	R:176 G:165 B:137	R:181 G:170 B:142	R:183 G:171 B:145	R:185 G:173 B:147	R:186 G:174 B:150	R:182 G:174 B:151	R:182 G:174 B:151	R:177 G:169 B:146	R:172 G:165 B:139
R:173 G:164 B: 133	R:183 G:172 B:144	R:188 G:177 B:149	R:188 G:176 B:150	R:188 G:176 B:152	R:188 G:176 B:152	R:180 G:172 B:151	R:180 G:172 B:151	R:175 G:167 B:144	R:171 G:163 B:140
R:180 G:171 B: 138	R:189 G:177 B:151	R:191 G:180 B:152	R:193 G:181 B:155	R:188 G:176 B:150	R:188 G:176 B:150	R:173 G:165 B:142	R:173 G:165 B:142	R:173 G:167 B:144	R:170 G:162 B:143
R:181 G:172 B: 139	R:188 G:176 B:150	R:190 G:179 B:151	R:190 G:178 B:152	R:190 G:178 B:152	R:185 G:173 B:147	R:178 G:166 B:144	R:170 G:162 B:141	R:167 G:159 B:140	R:161 G:153 B:134

Gambar 3.5 Nilai Pixel Cintra Background RGB

Untuk merubah menjadi grayscale nilai-nilai tersebut dihitung menggunakan persamaan (2.7) sebagai berikut :

Y(0,0) = (0,299*184)+(0,587*175)+(0,114*146) = 174,3 ~ 174

Y(0,1) = (0,299*182)+(0,587*171)+(0,114*143) = 171,0 ~ 171

Y(0,2) = (0,299*177)+(0,587*166)+(0,114*138) = 166,0 ~ 166

Y(0,3) = (0,299*172)+(0,587*161)+(0,114*133) = 161,0 ~ 161

Y(0,4) = (0,299*172)+(0,587*161)+(0,114*133) = 161,0 ~ 161

Y(0,5) = (0,299*177)+(0,587*165)+(0,114*139) = 165,6 ~ 166

Y(0,6) = (0,299*183)+(0,587*171)+(0,114*145) = 171,6 ~ 172

Y(0,7) = (0,299*181)+(0,587*175)+(0,114*143) = 173,1 ~ 173

Y(0,8) = (0,299*188)+(0,587*182)+(0,114*150) = 180,1 ~ 180

Y(0,9) = (0,299*189)+(0,587*183)+(0,114*149) = 180,9 ~ 181

Nilai Y(x,y) akan menggantikan 3 nilai piksel RGB menjadi 1 nilai grayscale sehingga untuk pixel pada koordinat (0,0) dengan nilai R=184, G=175, B=176 digantikan dengan nilai pixel Y(0,0) yaitu 174. Begitu pula untuk nilai piksel berikutnya hingga pixel terakhir Y(9,9).

Dengan perhitungan yang sama untuk setiap nilai pixel dalam matriks maka akan menghasilkan matriks citra grayscale sebagai berikut :

174	171	166	161	161	166	172	173	180	181
170	166	162	161	164	169	173	175	179	179
166	161	161	163	168	172	175	178	179	176
163	159	161	167	174	178	179	180	177	173
159	157	161	169	176	178	178	179	174	169
157	159	164	169	173	174	174	178	172	166
159	165	170	172	174	175	176	174	169	164
163	172	177	177	177	177	179	172	167	163
170	178	180	182	182	177	170	165	165	162
171	177	179	179	179	174	167	162	159	153

Gambar 3.6 Nilai Piksel Citra Background Hasil Grayscale

Setelah perhitungan konversi dari RGB ke grayscale, maka didapatkan citra background hasil grayscale sebagai berikut :



Gambar 3.7 Citra Background Hasil Greyscale

Pada citra hasil dari video frame didapat citra berikut :



Gambar 3.8 Citra dari Video Frame

Pada citra hasil dari video frame ini diambil citra ukuran 10x10 pixel pada suatu titik yang sama dengan permisalan pada citra background.

Nilai citra 10x10 dari hasil video frame adalah sebagai berikut :

R : 122 G : 123 B : 125	R : 114 G : 114 B : 112	R : 104 G :105 B : 99	R :111 G : 108 B : 99	R :149 G : 147 B : 134	R :124 G : 121 B : 106	R :128 G : 123 B : 103	R :172 G : 165 B : 137	R : 187 G : 180 B : 151	R :192 G : 185 B : 156
R : 102 G : 103 B : 107	R : 113 G : 113 B : 113	R : 119 G : 118 B : 114	R : 163 G : 160 B : 151	R : 116 G : 164 B : 151	R : 140 G : 137 B : 112	R :128 G : 123 B : 103	R : 167 G : 162 B : 133	R :183 G :177 B : 133	R :188 G : 181 B : 152
R : 84 G : 85 B : 87	R : 129 G : 129 B : 127	R : 108 G : 107 B : 102	R : 151 G : 148 B : 139	R : 130 G : 128 B : 115	R : 135 G : 132 B : 117	R : 121 G : 116 B : 96	R : 165 G : 159 B : 133	R : 165 G : 159 B : 133	R : 185 G : 178 B : 150
R : 136 G : 136 B : 138	R : 179 G : 178 B : 176	R : 104 G : 103 B : 98	R : 119 G : 116 B : 107	R : 112 G : 118 B : 106	R : 150 G : 114 B : 130	R : 126 G : 119 B : 100	R : 169 G : 162 B : 136	R : 169 G : 162 B : 136	R : 184 G : 177 B : 149
R : 204 G : 204 B : 204	R : 198 G : 197 B : 193	R : 115 G : 112 B : 105	R : 121 G : 117 B : 106	R : 151 G : 145 B : 131	R : 152 G : 146 B : 130	R : 115 G : 108 B : 89	R : 171 G : 164 B : 138	R : 171 G : 164 B : 138	R : 182 G : 175 B : 149
R : 221 G : 221 B : 219	R : 185 G : 184 B : 179	R : 127 G : 124 B : 117	R : 132 G : 125 B : 115	R : 145 G : 139 B : 125	R : 112 G : 115 B : 99	R : 113 G : 106 B : 87	R : 171 G : 163 B : 140	R : 171 G : 163 B : 140	R : 176 G : 169 B : 143
R : 203 G : 202 B : 197	R : 176 G : 173 B : 166	R : 130 G : 126 B : 117	R : 141 G : 135 B : 123	R : 131 G : 123 B : 110	R : 111 G : 102 B : 87	R : 145 G : 137 B : 118	R : 172 G : 164 B : 141	R : 172 G : 164 B : 141	R : 171 G : 163 B : 140
R : 169 G : 169 B : 161	R : 150 G : 146 B : 137	R : 104 G : 97 B : 87	R : 139 G : 131 B : 118	R : 128 G : 119 B : 104	R : 115 G : 106 B : 89	R : 158 G : 150 B : 131	R : 175 G : 167 B : 146	R : 175 G : 167 B : 146	R : 171 G : 163 B : 140
R : 133 G : 133 B : 121	R : 120 G : 116 B : 104	R :106 G : 100 B : 86	R :112 G : 105 B : 89	R : 137 G : 128 B : 111	R : 160 G : 148 B : 132	R :167 G : 156 B : 138	R :164 G : 156 B : 135	R :165 G : 157 B : 136	R :164 G : 156 B : 135
R : 119 G : 117 B : 104	R : 112 G : 106 B : 92	R : 111 G : 104 B : 88	R : 130 G : 121 B : 104	R : 160 G : 149 B : 131	R : 175 G : 162 B : 145	R : 174 G : 161 B : 144	R : 164 G : 153 B : 135	R : 160 G : 152 B : 133	R : 156 G : 148 B : 129

Gambar 3.9 Nilai Piksel Citra Frame RGB

Untuk merubah menjadi grayscale nilai-nilai tersebut dihitung menggunakan persamaan (2.7) sebagai berikut :

Y(0,0) = (0,299*122)+(0,587*123)+(0,114*125) = 122,9 ~ 123

Y(0,1) = (0,299*114)+(0,587*114)+(0,114*112) = 113,7 ~ 123

Y(0,2) = (0,299*104)+(0,587*105)+(0,114*99) = 104,0 ~ 104

Y(0,3) = (0,299*111)+(0,587*108)+(0,114*99) = 107,8 ~ 108

Y(0,4) = (0,299*149)+(0,587*147)+(0,114*134) = 146,1 ~ 146

Y(0,5) = (0,299*124)+(0,587*121)+(0,114*106) = 120,1 ~ 120

Y(0,6) = (0,299*128)+(0,587*123)+(0,114*103) = 122,2 ~ 122

Y(0,7) = (0,299*172)+(0,587*165)+(0,114*137) = 163,9 ~ 164

Y(0,8) = (0,299*187)+(0,587*180)+(0,114*151) = 178,7 ~ 179

Y(0,9) = (0,299*192)+(0,587*185)+(0,114*156) = 183,7 ~ 184

Nilai Y(x,y) akan menggantikan 3 nilai piksel RGB menjadi 1 nilai grayscale sehingga untuk pixel pada koordinat (0,0) dengan nilai R=122, G=123, B=125 digantikan dengan nilai pixel Y(0,0) yaitu 123. Begitu pula untuk nilai piksel berikutnya hingga pixel terakhir Y(9,9).

Dengan perhitungan yang sama untuk setiap nilai pixel dalam matriks maka akan menghasilkan matriks citra grayscale sebagai berikut :

123	114	104	108	146	120	122	164	179	184
103	113	118	160	163	136	122	160	175	180
85	129	107	148	127	131	115	160	172	177
136	178	103	116	118	144	119	158	172	176
204	197	112	117	145	146	108	161	171	174
221	184	124	126	139	115	106	163	168	168
202	173	126	135	124	103	137	164	166	163
168	146	98	132	120	107	150	167	167	163
132	116	100	105	129	150	157	156	157	156
116	106	104	122	150	164	163	154	152	148

Gambar 3.10 Nilai Piksel Citra Frame Hasil Grayscale

Setelah perhitungan konversi dari RGB ke grayscale, maka didapatkan citra video frame hasil grayscale sebagai berikut :



Gambar 3.11 Citra Video Frame Hasil Grayscale

3.3.3 Background Subtraction

Setelah data inputan berupa video rekaman dan citra background melalui proses ekstraksi video frames dan grayscale maka akan didapatkan kumpulan video frames dan background image. Baik video frames atau background image telah melalui proses grayscale sehingga citra yang dihasilkan merupakan citra grayscale. Selanjutnya, citra video frames dan background akan dilakukan proses background subtraction dan thresholding untuk mendeteksi objek.

3.3.3.1 Background Subtraction (Foreground Detection)

Background subtraction atau Foreground Detection merupakan proses untuk mendeteksi pergerakan atau perbedaan yang signifikan dari video frame. Frame video akan dibandingkan dengan background images yang ada pada tahap sebelumnya. Proses ini akan mengurangkan nilai matrix citra current frame dan nilai matrix citra background menggunakan perhitungan *absolut difference*.

Perhitungan absolut difference dapat dilakukan dengan persamaan (8) sebagai berikut :

$$Z_{(x,y)} = \left| B_{(x,y)} - C_{(x,y)} \right|$$

dimana :

Z = nilai piksel absolute

B = nilai piksel background

C = nilai piksel curent frame

(x,y) = koordinat nilai piksel pada matriks

Nilai piksel hasil grayscale digunakan kembali pada proses ini. Nilai piksel background adalah sebagai berikut :

174	171	166	161	161	166	172	173	180	181
170	166	162	161	164	169	173	175	179	179
166	161	161	163	168	172	175	178	179	176
163	159	161	167	174	178	179	180	177	173
159	157	161	169	176	178	178	179	174	169
157	159	164	169	173	174	174	178	172	166
159	165	170	172	174	175	176	174	169	164
163	172	177	177	177	177	179	172	167	163
170	178	180	182	182	177	170	165	165	162
171	177	179	179	179	174	167	162	159	153

Gambar 3.12 Nilai Piksel Citra Background Hasil Grayscale

Sedangkan nilai piksel video frame adalah sebagai berikut :

123	114	104	108	146	120	122	164	179	184
103	113	118	160	163	136	122	160	175	180
85	129	107	148	127	131	115	160	172	177
136	178	103	116	118	144	119	158	172	176
204	197	112	117	145	146	108	161	171	174
221	184	124	126	139	115	106	163	168	168
202	173	126	135	124	103	137	164	166	163
168	146	98	132	120	107	150	167	167	163
132	116	100	105	129	150	157	156	157	156
116	106	104	122	150	164	163	154	152	148

Gambar 3.13 Nilai Piksel Citra Frame Hasil Grayscale

Setiap nilai piksel background dan video frame dihitung dengan persamaan (8) agar diketahui selisih nilai piksel absolut dari kedua citra. Kumpulan tersebut merupakan objek yang dapat disebut sebagai foreground mask. Perhitungannya adalah sebagai berikut :

$Z(0,0) = | 174 – 123 | = 51$

$Z(0,1) = | 171 – 114 | = 57$

$Z(0,2) = | 166 – 104 | = 72$

$Z(0,3) = | 161 – 108 | = 53$

$Z(0,4) = | 161 – 146 | = 15$

$Z(0,5) = | 166 – 120 | = 46$

$Z(0,6) = | 172 – 122 | = 50$

$Z(0,7) = | 173 – 164 | = 9$

Z(0,8) = | 180 – 179 | = 1

Z(0,9) = | 181 – 184 | = 3

Perhitungan tersebut dilakukan pada setiap nilai pixel sebagai berikut :

|174-123|	|171-114|	|166-104|	|161-108|	|161-146|	|166-120|	|172-122|	|173-164|	|180-179|	|181-184|
|170-103|	|166-113|	|162-118|	|161-160|	|164-163|	|169-136|	|173-122|	|175-160|	|179-175|	|179-180|
|166-85|	|161-129|	|161-107|	|163-148|	|168-127|	|172-131|	|175-115|	|178-160|	|179-172|	|176-117|
|163-136|	|159-178|	|161103-|	|167-116|	|174-118|	|178-144|	|179-119|	|180-158|	|177-172|	|173-176|
|159-204|	|157-197|	|161-112|	|169-117|	|176-145|	|178-146|	|178-108|	|179-161|	|174-171|	|169-174|
|157-221|	|159-184|	|164-124|	|169-126|	|173-139|	|174-115|	|174-106|	|178-163|	|172-168|	|166-168|
|159-202|	|165-173|	|170-126|	|172-135|	|174-124|	|175-103|	|176-137|	|174-164|	|169-166|	|164-163|
|163-168|	|172-146|	|177-98|	|177-132|	|177-120|	|177-107|	|179-150|	|172-167|	|167-167|	|163-163|
|170-132|	|178-116|	|180-100|	|182-105|	|182-129|	|177-150|	|170-157|	|165-156|	|165-157|	|162-156|
|171-116|	|177-106|	|179-104|	|179-122|	|179-150|	|174-164|	|167-163|	|162-154|	|159-152|	|153-148|

Gambar 3.14 Perhitungan Nilai Absolut Diference

Dari perhitungan tersebut diperoleh matriks hasil pengurangan atau matriks absoulut difference sebagai berikut :

51	57	72	53	15	46	50	9	1	3
67	53	44	1	1	33	33	51	4	1
81	32	54	15	41	41	60	18	7	59
27	19	58	1	56	34	60	22	5	3
45	40	49	52	31	32	70	18	3	5
54	25	40	43	34	59	68	15	4	2
43	8	44	37	50	72	39	10	3	1
5	26	79	45	57	70	29	5	0	0
38	62	80	77	53	27	13	9	8	6
55	71	75	57	29	10	4	8	7	5

Gambar 3.15 Nilai Piksel Hasil Absolute Difference

Setelah perhitungan background subtraction, maka didapatkan citra hasil background subtraction sebagai berikut :



Gambar 3.16 Citra Hasil Background Subtraction

3.3.3.2 Threshold

Setelah melakukan background subtraction selanjutnya dilakukan thresholding. Thresholding merupakan teknik yang sederhana dan efektif untuk segmentasi citra. Proses thresholding sering disebut dengan proses binerisasi. Pada beberapa aplikasi pengolahan citra, terlebih dahulu dilakukan threshold terhadap citra gray level untuk dapat menjadi citra biner (citra yang memiliki nilai level keabuan 0 atau 255). Sebuah citra hasil proses thresholding dapat disajikan dalam histogram citra untuk mengetahui penyebaran nilai-nilai intensitas piksel pada suatu citra/bagian tertentu dalam citra sehingga untuk citra bimodal, histogram dapat dipartisi dengan baik (segmentasi objek dengan background) dan dapat ditentukan nilai threshold-nya.

Rumus untuk menentukan nilai Threshold bisa didapatkan dari persamaan (9) sebagai berikut:

$$g(x,y) = \{bernilai\ 1\ jika\ f(x,y) \geq\ T\}$$
$$g(x,y) = \{bernilai\ 0\ jika\ f(x,y) < T\}$$

dimana :

g(x,y) = nilai matrix citra hasil thresholding.
f(x,y) = merupakan nilai matrix citra yang akan di-threshold.
T = merupakan nilai threshold (0 – 255)

Nilai threshold bergantung pada kebutuhan. Pada citra hasil background subtraction selanjutnya akan dilakukan thresholding dengan nilai T=30 sehingga proses perhitungannya adalah sebagai berikut :

51	57	72	53	15	46	50	9	1	3
67	53	44	1	1	33	33	51	4	1
81	32	54	15	41	41	60	18	7	59
27	19	58	1	56	34	60	22	5	3
45	40	49	52	31	32	70	18	3	5
54	25	40	43	34	59	68	15	4	2
43	8	44	37	50	72	39	10	3	1
5	26	79	45	57	70	29	5	0	0
38	62	80	77	53	27	13	9	8	6
55	71	75	57	29	10	4	8	7	5

Gambar 3.17 Nilai Piksel Hasil Perhitungan Absolute Difference

Untuk matriks (0,0)
f(0,0) = 51
g(0,0) = {51 > 30}
g(0,0) = 1

Untuk matriks (0,1)
f(0,0) = 57
g(0,0) = {57 > 30}
g(0,0) = 1

Untuk matriks (0,2)
f(0,0) = 72
g(0,0) = {72 > 30}
g(0,0) = 1

Untuk matriks (0,3)
f(0,0) = 53
g(0,0) = {53 > 30}
g(0,0) = 1

Untuk matriks (0,4)
f(0,0) = 15
g(0,0) = {15 < 30}
g(0,0) = 0

Untuk matriks (0,5)
f(0,0) = 46
g(0,0) = {46 > 30}
g(0,0) = 1

Untuk matriks (0,6)
f(0,0) = 50
g(0,0) = {50 > 30}
g(0,0) = 1

Untuk matriks (0,7)
f(0,0) = 9
g(0,0) = {9 < 30}
g(0,0) = 0

Untuk matriks (0,8)
f(0,0) = 1
g(0,0) = {1 < 30}
g(0,0) = 0

Untuk matriks (0,9)
f(0,0) = 3
g(0,0) = {3 < 30}
g(0,0) = 0

Dengan perhitungan yang sama untuk setiap nilai pixel dalam matriks dengan nilai T=30 maka akan menghasilkan matriks hasil threshold sebagai berikut :

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

Gambar 3.18 Nilai Piksel Citra Hasil Threshold

Setelah dilakukan thershold dengan nilai T=30, maka didapatkan citra hasil threshold sebagai berikut :



Gambar 3.19 Citra Hasil Threshold

3.3.4 Pemberian Batas Jalur

Citra hasil threshold menghasilkan citra dengan nilai pixel biner (0 atau 1). Pada proses perhitungan jumlah mobil dapat dilakukan dengan memberikan batas jalur pada citra. Pemberian batas ini dilakukan dengan membuat objek persegi pada citra untuk selanjutnya dihitung jumlah pixel didalam persegi tersebut.

Berdasarkan perhitungan kepadatan pada pembahasan sebelumnya, yang menyebutkan bahwa apabila setiap satu loket pembayaran gerbang tol terdiri dari 1 mobil dilayani dan 5 mobil mengantri maka dapat dikatakan gerbang tol yang diamati tersebut sedang mengalami kepadatan, maka pemberian batas jalur ini dilakukan dengan membuat 4 kotak yang selanjutnya disebut dengan Stage (stage 1, stage 2, stage 3, stage 4) untuk mengetahui jumlah mobil pada masing masing jalur. Jalur yang diamati terdiri dari 3 jalur. Sehingga pemberian kotak pembatas jalur dibuat 12 kotak (4 kotak x 3 jalur). Objek persegi digunakan untuk menghitung pixel berwarna putih atau bernilai 1 yang menandakan bahwa ada objek yang terdeteksi pada kotak tersebut.

Pada citra hasil background subtraction dan threshold yang ada terdapat pada Gambar 3.8, pemberian jalur dilakukan mulai dari bagian bawah citra yang

artinya mendeteksi mobil pada baris paling bawah dari video lalu kebagian atas secara berurutan. Pemberian kotak jalur ini kemudian disebut dengan kotak Stage 1, Stage 2, Stage 3, dan Stage 4.

1. Stage 1

Pemberian batas jalur stage 1 ini merupakan batas untuk menghitung objek mobil yang terdeteksi pada baris paling bawah video yang artinya stage 1 ini berperan untuk menghitung jumlah mobil yang sedang mengantri paling depan.

Berdasarkan ukuran kendaraan pada tabel 2.2, ukuran kendaraan terbesar adalah 2x13 meter. Pada hasil rekaman video, mobil dengan ukuaran paling besar terekam pada posisi terdepan dengan ukuran citra rata-rata 55x56. Video hasil rekaman memiliki ukuran. Sehingga berdasarkan ukuran tersebut, pada citra hasil background subtraction dan threshold, pemberian batas jalur stage 1 ditempatkan pada posisi sebagai berikut :

Titik Koordinat Kotak Stage 1 jalur ke-1

Koordinat kotak : (61,211) (116,211);(61,267) ;(116,267)

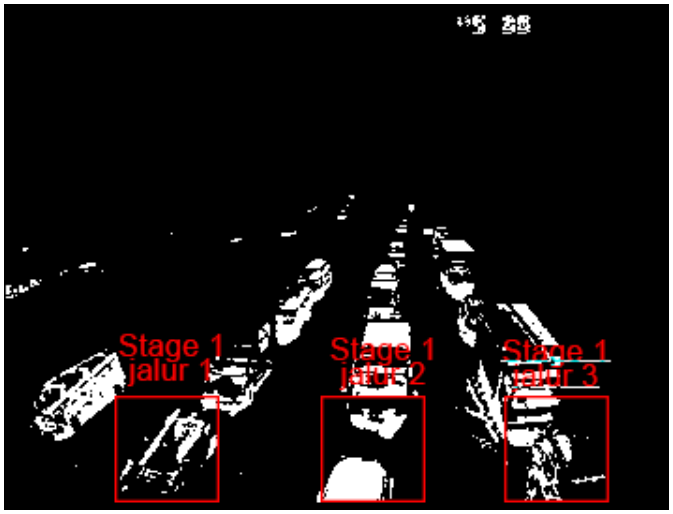
Titik Koordinat Kotak Stage 1 jalur ke-2

Koordinat kotak : (171,211) ;(226,211);(171,267);(226,267)

Titik Koordinat Kotak Stage 1 jalur ke-3

Koordinat kotak : (269,211) ;(324,211);(269,267);(324,267)

Kotak-kotak tersebut ditunjukan pada posisi seperti gambar berikut :



Gambar 3.20 Pemberian Batas Jalur Stage 1

2. Stage 2

Pemberian batas jalur stage 2 ini merupakan batas untuk menghitung objek mobil yang terdeteksi pada baris kedua yang artinya stage 2 ini berperan untuk menghitung jumlah mobil yang sedang mengantri pada barisan kedua.

Berdasarkan ukuran kendaraan pada tabel 2.2, ukuran kendaraan terbesar adalah 2x13 meter. Pada hasil rekaman video, mobil dengan ukuaran paling besar terekam pada posisi terdepan dengan ukuran citra rata-rata pada 42x40. Pada citra hasil background subtraction dan threshold, pemberian batas jalur stage 2 ditempatkan pada posisi sebagai berikut :

Titik Koordinat Kotak Stage 2 jalur ke-1

Koordinat kotak : (101,170);(143,170);(101,210);(143,210)

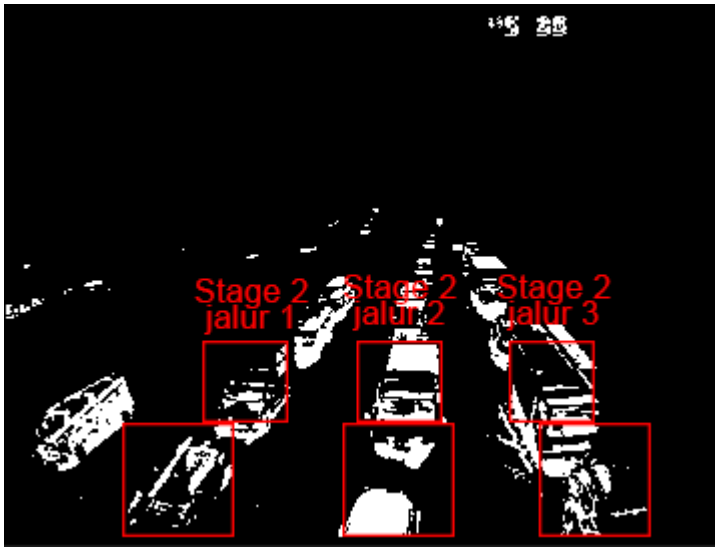
Titik Koordinat Kotak Stage 2 jalur ke-2

Koordinat kotak : (178,170);(220,170);(178,210);(220,210)

Titik Koordinat Kotak Stage 2 jalur ke-3

Koordinat kotak : (254,170);(296,170);(254,210);(296,210)

Kotak-kotak tersebut ditunjukan pada posisi seperti gambar berikut :



Gambar 3.21 Pemberian Batas Jalur Stage 2

3. Stage 3

Pemberian batas jalur stage 3 ini merupakan batas untuk menghitung objek mobil yang terdeteksi pada baris ketiga yang artinya stage 3 ini berperan untuk menghitung jumlah mobil yang sedang mengantri pada barisan ketiga.

Berdasarkan ukuran kendaraan pada tabel 2.2, ukuran kendaraan terbesar adalah 2x13 meter. Pada hasil rekaman video, mobil dengan ukuaran paling besar terekam pada posisi terdepan dengan ukuran citra rata-rata pada 26x23. Pada citra hasil background subtraction dan threshold, pemberian batas jalur stage 3 ditempatkan pada posisi sebagai berikut :

Titik Koordinat Kotak Stage 3 jalur ke-1

Koordinat kotak : (139,146); (165,146);(139,169);(165,169)

Titik Koordinat Kotak Stage 3 jalur ke-2

Koordinat kotak : (186,146);(212,146);(186,169);(212,169)

Titik Koordinat Kotak Stage 3 jalur ke-3

Koordinat kotak : (238,146);(264,146);(238,169);(264,169)

Kotak-kotak tersebut ditunjukan pada posisi seperti gambar berikut :



Gambar 3.22 Pemberian Batas Jalur Stage 3

4. Stage 4

Pemberian batas jalur stage 4 ini merupakan batas untuk menghitung objek mobil yang terdeteksi pada baris keempat yang artinya stage 4 ini berperan untuk menghitung jumlah mobil yang sedang mengantri pada barisan keempat.

Berdasarkan ukuran kendaraan pada tabel 2.2, ukuran kendaraan terbesar adalah 2x13 meter. Pada hasil rekaman video, mobil dengan ukuaran paling besar terekam pada posisi terdepan dengan ukuran citra rata-rata pada 18x16. Pada citra hasil background subtraction dan threshold, pemberian batas jalur stage 4 ditempatkan pada posisi sebagai berikut :

Titik Koordinat Kotak Stage 4 jalur ke-1

Koordinat kotak : (159,129);(177,129);(159,145);(177,145)

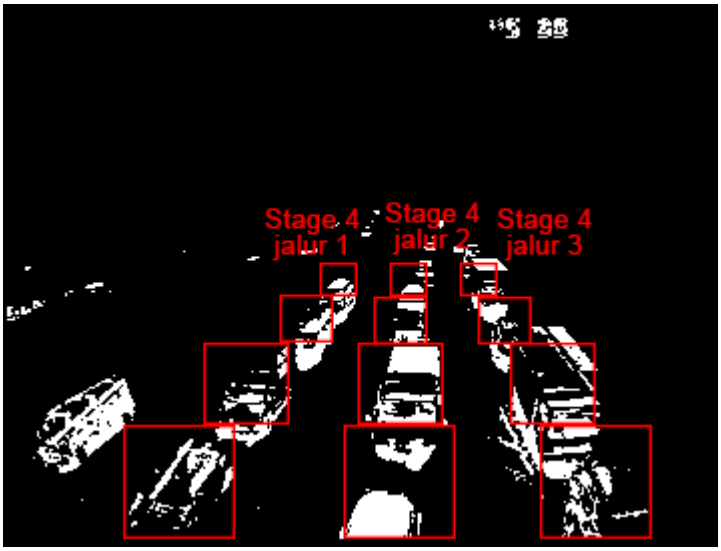
Titik Koordinat Kotak Stage 4 jalur ke-2

Koordinat kotak : (194,129);(212,129);(194,145);(212,145)

Titik Koordinat Kotak Stage 4 jalur ke-3

Koordinat kotak : (229,129);(247,129);(229,145);(247,145)

Kotak-kotak tersebut ditunjukan pada posisi seperti gambar berikut :



Gambar 3.23 Pemberian Batas Jalur Stage 4

3.3.5 Penentuan Kepadatan

Berdasarkan perhitungan kepadatan pada pembahasan sebelumnya, yang menyebutkan bahwa apabila setiap satu loket pembayaran gerbang tol terdiri dari 1 mobil dilayani dan 5 mobil mengantri maka dapat dikatakan gerbang tol yang diamati tersebut sedang mengalami kepadatan. Maka dalam penentuan kepadatan dibutuhkan perhitungan objek mobil dan menghitung apakah mobil sudah terdiri dari 5 antrian untuk setiap jalur. Sebelum melakukan penentuan kepadatan terlebih dahulu dilakukan perhitungan jumlah objek mobil yang terdeteksi.

3.3.5.1 Perhitungan Jumlah Objek Mobil

Dengan memanfaatkan kotak pembatas jalur yang telah dilakukan sebelumnya, perhitungan objek mobil dapat dilakukan dengan menghitung setiap pixel putih atau yang bernilai 1 pada citra. Untuk menentukan apakah objek dalam kotak tersebut merupakan objek mobil atau bukan makan dibutuhkan ambang batas jumlah piksel 1 pada setiap kotak yang ada pada setiap stage.

Setiap stage diberikan ambang batas untuk menentukan objek terdeteksi merupakan mobil, ambang batas diberikan sebagai berikut :

1. Stage 1

   Kotak stage 1 memilik ukuran 55x56 yang artinya didalam kotak tersebut terdapat 3080 piksel. Berdasarkan pengamatan pada video dan ukuran mobil terkecil seperti pada tabel 2.2 maka ambang batas untuk menentukan objek mobil pada stage ini adalah antara 325-429 piksel.

2. Stage 2

   Kotak stage 2 memilik ukuran 42x40 yang artinya didalam kotak tersebut terdapat 1680 piksel. Berdasarkan pengamatan pada video dan ukuran mobil terkecil seperti pada tabel 2.2 maka ambang batas untuk menentukan objek mobil pada stage ini adalah 162-245 piksel.

3. Stage 3

   Kotak stage 3 memilik ukuran 26x23 yang artinya didalam kotak tersebut terdapat 598 piksel. Berdasarkan pengamatan pada video dan ukuran mobil terkecil seperti pada tabel 2.2 maka ambang batas untuk menentukan objek mobil pada stage ini adalah 104-144 piksel.

4. Stage 4

   Kotak stage 1 memilik ukuran 18x16 yang artinya didalam kotak tersebut terdapat 288 piksel. Berdasarkan pengamatan pada video dan ukuran mobil terkecil seperti pada tabel 2.2 maka ambang batas untuk menentukan objek mobil pada stage ini adalah 20-57 piksel.

Pemberian ambang batas nilai piksel tersebut selanjutnya akan digunakan untuk menentukan dan menghitung objek mobil yang terdapat pada citra hasil background dan threshold.

Jika jumlah piksel putih pada kotak setiap stage lebih dari atau sama dengan ambang batas maka pada kotak stage tersebut terdapat objek mobil.

Misalnya pada citra stage 1 dilakukan perhitungan jumlah piksel 1 pada setiap kotak jalur 1, jalur 2 dan jalur 3



Gambar 3.24 Perhitungan Jumlah Objek Mobil Stage 1

Pada kotak jalur 1, terdeteksi sebanyak 490 piksel putih

490 > 429 = Ya (terdapat objek mobil)

Pada kotak jalur 2, terdeteksi sebanyak 672 piksel putih

672 > 429 = Ya (terdapat objek mobil)

Pada kotak jalur 2, terdeteksi sebanyak 464 piksel putih

464 > 429 = Ya (terdapat objek mobil)

Maka hasil pada kotak stage 1 dapat disimpulkan bahwa terdapat 3 mobil yang terdeteksi.

Proses ini dilakukan untuk setiap kotak yang terdapat pada setiap stage sehingga akan diketahui jumlah objek mobil yang terdeteksi.

3.3.5.2 Klasifikasi Traffic

Klasifikasi traffic merupakan proses untuk menentukan apakah lalu lintas gerbang tol sedang pada atau tidak. Parameter dalam menentukan klasifikasi ini adalah jumlah mobil yang terdeteksi.

Berdasarkan pembahasan teori antrian, gerbang tol pasteur memiliki tingkat intensitas fasilitas pelayanan sebesar 34,07% yang menyimpulkan bahwa pada sistem antrian gerbang tol akan selalu terjadi antrian. Pada analisis kepadatan gerbang tol menyimpulkan bahwa apabila setiap satu loket pembayaran gerbang tol terdiri dari 1 mobil dilayani dan 5 mobil mengantri maka dapat dikatakan gerbang tol yang diamati tersebut sedang mengalami kepadatan. Sehingga dalam penentuan kepadatan gerbang tol ini didasarkan pada jumlah objek mobil yang sedang mengantri.

Pada pemberian batas jalur yang telah dibahas sebelumnya, terdapat 4 stage dengan jumlah kotak setiap stage 3, sehingga terdapat 12 kotak untuk mendeteksi 12 objek mobil.

Terdapat 2 klasifikasi yaitu :

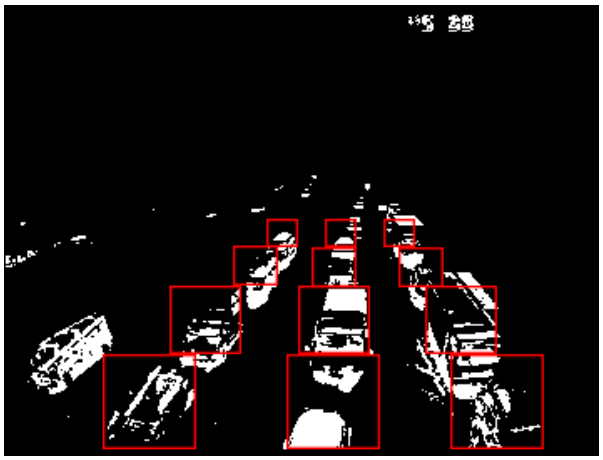
1. Padat

   Pengklasifikasian dikatakan padat apabila setiap kotak pada setiap stage mendeteksi adanya objek mobil. Artinya apabila pada proses perhitungan jumlah objek mobil terdapat 12 mobil terdeteksi maka dikatakan padat.

2. Tidak Padat

   Pengklasifikasian dikatakan Tidak Padat apabila hanya sebagian kotak pada setiap stage mendeteksi adanya objek mobil. Artinya apabila pada proses perhitungan jumlah objek mobil terdapat kurang dari 12 mobil terdeteksi maka dikatakan tidak padat.

Misalkan pada citra hasil background subtraction, threshold dan pemberian batas jalur sebagai berikut :



Gambar 3.25 Citra Hasil Pemberian Jalur

Pada kotak stage 1 mendeteksi terdapat 3 objek mobil.

Pada kotak stage 2 mendeteksi terdapat 3 objek mobil.

Pada kotak stage 3 mendeteksi terdapat 3 objek mobil.

Pada kotak stage 4 mendeteksi terdapat 3 objek mobil.

Jumlah mobil terdeteksi = 3+3+3+3 = 12 objek mobil maka pada proses klasifikasi, kondisi tersebut dikatakan sebagai “Padat”.

3.4 Analisis Non Fungsional

Kebutuhan non fungsional yang dibutuhkan untuk membangun sistem ini mencakup 2 hal yaitu kebutuhan perangkat keras dan perangkat lunak.

3.4.1 Analisis Kebutuhan Perangkat Keras

Adapun kebutuhan perangkat keras yang diperlukan untuk menjalankan sistem ini tertera dalam tabel dibawah ini :

Tabel 3.4 Kebutuhan Perangkat Keras

No	Nama Perangkat	Spesifikasi
1	Processor	Intel Core 2 Duo @2,3Ghz
2	Memory	2 GB
3	Harddisk	Minimal 300 MB
4	Monitor	14 inch Resolusi 1024 x 640
5	Keyboard & Mouse	

3.4.2 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk menjalankan aplikasi ini adalah:

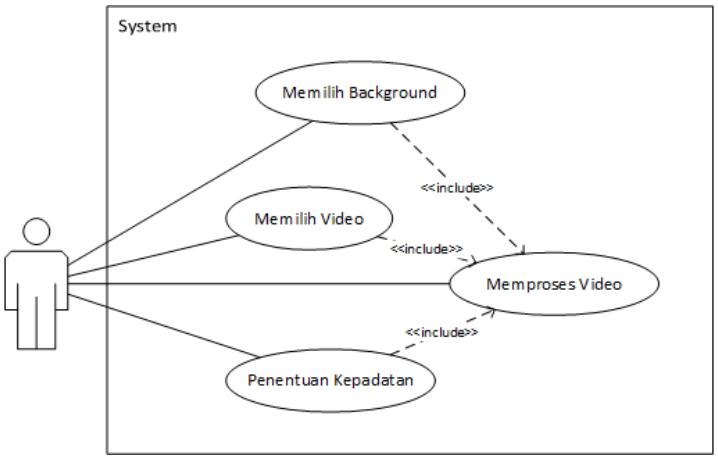
1. Sistem Operasi Windows 7 atau yang terbaru.
2. Netbeans IDE 8.0.2 atau yang terbaru.
3. Java Development Kit JDK versi 8 atau yang terbaru.

3.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk menguraikan proses kegiatan dan kebutuhan yang akan diterapkan pada sistem yang akan dibangun. Kebutuhan fungsionalitas pada aplikasi yang akan dibangun terdiri dari use case diagram, activity diagram, sequence diagram, dan class diagram.

3.5.1 Use Case Diagram

Use case diagram merupakan gambaran fungsionalitas atau perilaku dari sistem. Berikut use case diagram pada aplikasi dapat dilihat pada gambar berikut :



Gambar 3.26 Use Case Diagram

Deskripsi pendefinisian aktor dan deskripsi pendefisian use case pada sistem dapat dilihat pada tabel tabel berikut :

Tabel 3.5 Definisi Aktor

No	Aktor	Deskripsi
1	User	Orang yang akan melakukan analisis video

Tabel 3.6 Definisi Use Case

No	Use Case	Deskripsi
1	Memasukkan Video	Fungsionalitas untuk memasukan data inputan berupa video kedalam system
2	Memasukkan Background	Fungsionalitas untuk melakukan registrasi background untuk kepentingan proses
3	Memproses Video	Fungsionalitas untuk memproses video dimulai dari pre-processing, background subtraction dan perhitungan teori antrian.
4	Penentuan Kepadatan	Fungsionalitas untuk menganalisis hasil deteksi background subtraction dan perhitungan jumlah kendaraan.

3.5.1.1 Skenario Use Case

Skenario use case mendeskripsikan langkah-langkah dalam proses sistem, baik dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Penjelasan skenario use case adalah sebagai berikut :

1. Skenario Use Case Memasukkan Video (*Input Video*)

Skenario use case input video dapat dilihat pada tabel berikut:

Tabel 3.7 Skenario Use Case Input Video

Identifikasi	
Nama	Input Video
Tujuan	Memastikan user untuk memasukkan video input yang sesuai baik video maupun formatnya.
Deskripsi	Menu ini akan menampilkan daftar video input yang ada pada direktori penyimpanan user.
Aktor	User
Skenario Utama	
Aksi Aktor	Reaksi Sistem
User mengklik tombol Input Video	Sistem akan menampilkan daftar video yang dapat dipilih
Skenario Alternatif – Input Video Gagal	
Aksi Aktor	Reaksi Sistem
User kembali mengklik tombol Input Video	Sistem akan kembali menampilkan daftar video yang dapat dipilih sehingga user memilih video yang sesuai dengan format yang dibutuhkan.
Kondisi Akhir	User dapat melakukan input video untuk nantinya dilanjutkan ke tahap pemrosesan video.

2. Skenario Use Case Input Background

Skenario use case input background dapat dilihat pada tabel berikut:

Tabel 3.8 Skenario Use Case Input Background

Identifikasi	
Nama	Input Background
Tujuan	Memastikan user untuk memasukkan background yang sesuai.
Deskripsi	Menu ini akan menampilkan daftar background yang ada pada direktori penyimpanan user.
Aktor	User
Skenario Utama	
Aksi Aktor	Reaksi Sistem

User mengklik tombol Input Video	Sistem akan menampilkan daftar background yang dapat dipilih.
Skenario Alternatif – Input Background Gagal	
Aksi Aktor	Reaksi Sistem
User kembali mengklik tombol Input Video	Sistem akan kembali menampilkan daftar background yang dapat dipilih sehingga user memilih background yang sesuai dengan format yang dibutuhkan.
Kondisi Akhir	User dapat melakukan input background untuk nantinya dilanjutkan ke tahap pemrosesan video.

3. Skenario Use Case Proses Video

Skenario use case proses video dapat dilihat pada tabel berikut

Tabel 3.9 Skenario Use Case Proses Video

Identifikasi	
Nama	Background Subtraction
Tujuan	Agar dapat melakukan pemrosesan video dan background sehingga sistem dapat menganalisa dan memberikan informasi mengenai hasil analisis dari video.
Deskripsi	Menu ini akan menampilkan proses analisis video.
Aktor	User
Skenario Utama	
Aksi Aktor	Reaksi Sistem
User mengklik tombol Proses Video	Sistem akan melakukan pemrosesan mulai dari pengecekan background, video, pre-processing, background subtraction, perhitungan objek mobil.
Skenario Alternatif – Proses Video Gagal	
Aksi Aktor	Reaksi Sistem
User kembali mengklik tombol Input Video dan Input Background	Sistem akan kembali menampilkan daftar background dan daftar video yang dapat dipilih sehingga user memilih background dan video yang sesuai dengan format yang dibutuhkan.
Kondisi Akhir	User dapat mendapat hasil analisis video yang dilakukan oleh system.

4. Skenario Use Case Penentuan Kepadatan

Skenario use case analisis antrian dapat dilihat pada tabel berikut

Tabel 3.10 Skenario Use Case Analisis Traffic

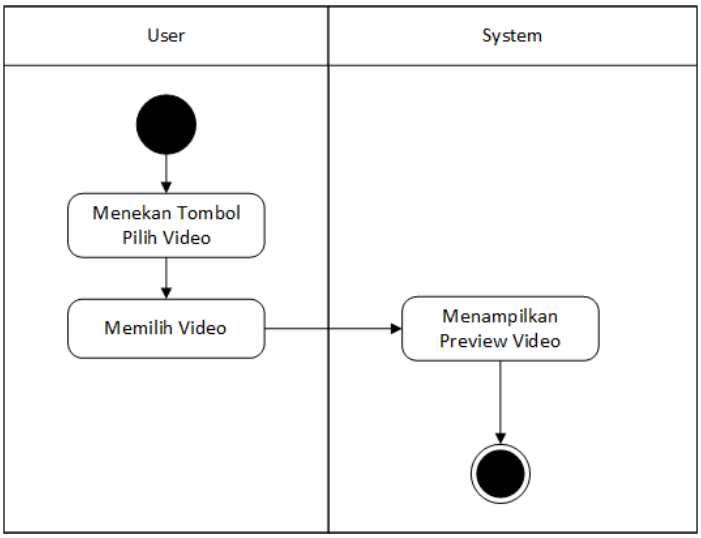
Identifikasi	
Nama	Penentuan Kepadatan
Tujuan	Menganalisis hasil pemrosesan video berdasarkan teori antrian untuk menentukan kepadatan gerbang tol
Deskripsi	Sistem akan menganalisis hasil pemrosesan video
Aktor	User
Skenario Utama	
Aksi Aktor	Reaksi Sistem
User mengklik tombol Proses Video	Sistem akan melakukan perhitungan antrian dan klasifikasi traffic
Skenario Alternatif – Proses Video Gagal	
Aksi Aktor	Reaksi Sistem
User kembali mengklik tombol Input Video dan Input Background	Sistem akan kembali menampilkan daftar background dan daftar video yang dapat dipilih sehingga user memilih background dan video yang sesuai dengan format yang dibutuhkan.
Kondisi Akhir	User dapat mendapat hasil analisis video yang dilakukan oleh system.

3.5.1.2 Activity Diagram

Activity diagram memodelkan aliran kerja dari urutan aktivitas dalam suatu proses yang mengacu pada use case diagram yang ada. Penjelasan masing-masing activity diagram adalah sebagai berikut :

1. Activity Diagram Input Video

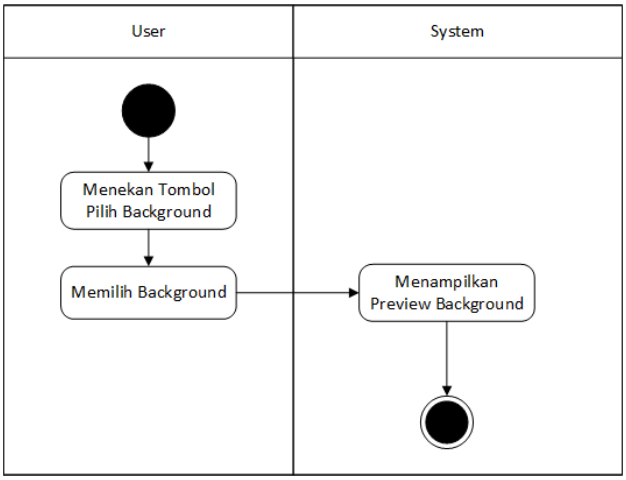
Activity diagram input video menjelaskan aliran kerja User memasukkan data berupa rekaman video kedalam sistem. Berikut activity diagram dari input video:



Gambar 3.27 Activity Diagram Input Video

2. Activity Diagram Input Background

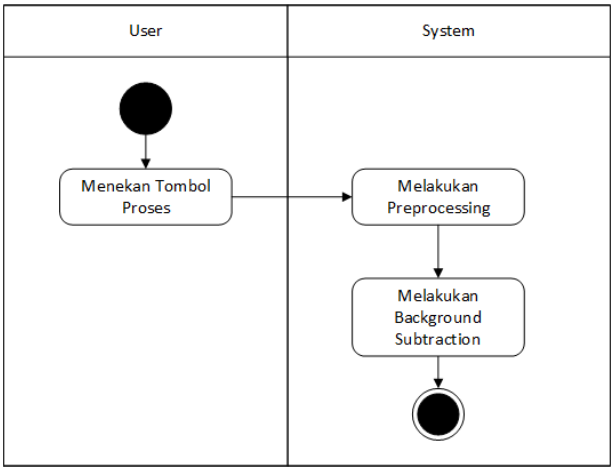
Activity diagram input background menjelaskan aliran kerja User memasukkan data berupa citra background yang sesuai dengan video kedalam sistem. Berikut activity diagram dari input background:



Gambar 3.28 Activity Diagram Input Background

3. Activity Diagram Proses Video

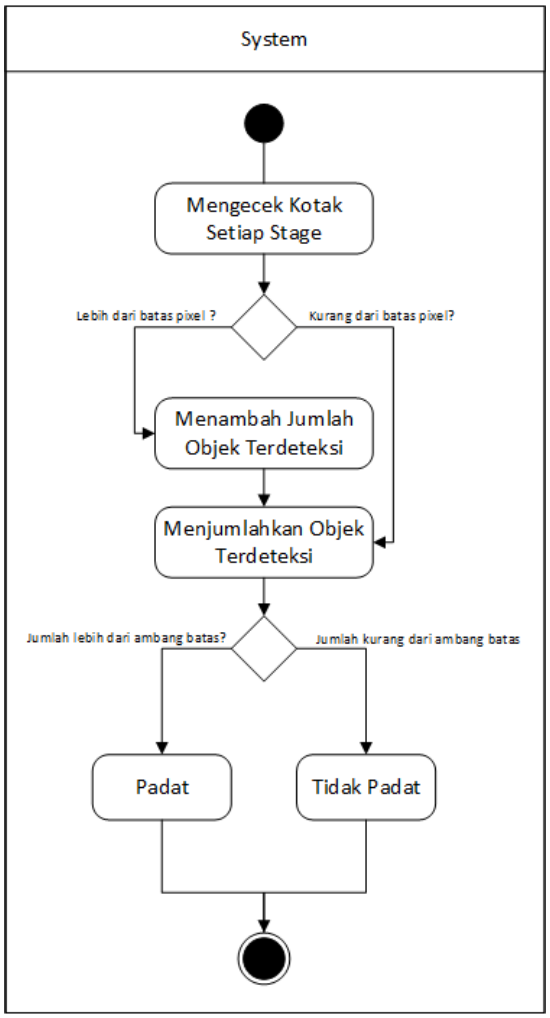
Activity diagram proses video menjelaskan aliran kerja pemrosesan video. Pemrosesan dimulai dengan mengubah video menjadi video frames, background subtraction, threshold dan perhitungan jumlah kendaraan yang terekam. Berikut activity diagram dari proses video:



Gambar 3.29 Activity Diagram Proses Video

4. Activity Diagram Penentuan Kepadatan

Activity diagram analisis antrian menjelaskakn aliran kerja analisis antrian dan klasifikasi traffic. Berikut activity diagram dari analisis antrian:



Gambar 3.30 Activity Diagram Analisis Traffic

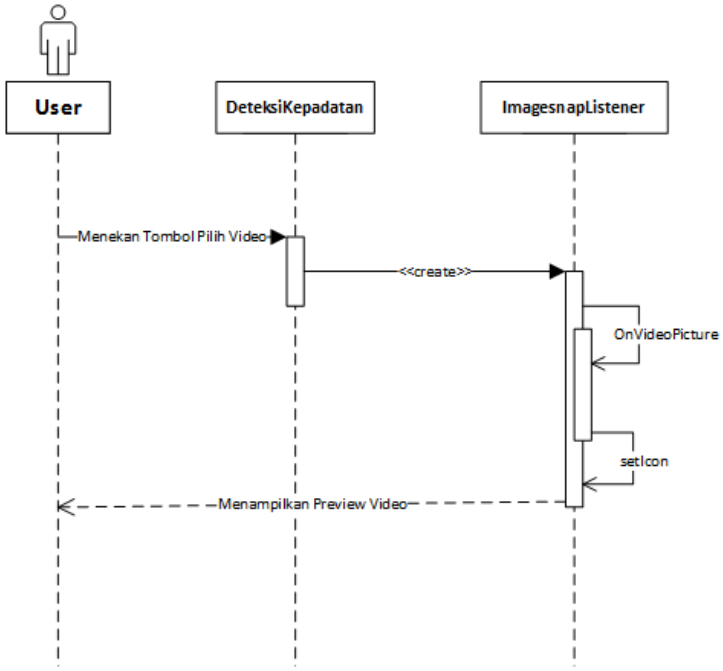
3.5.2 Sequence Diagram

Sequence diagram adalah suatu diagram interaksi yang menekankan pada pengaturan waktu dari pesan-pesan. Diagram ini menampilkan sekumpulan peran dan pesan-pesan yang dikirim dan diterima oleh instansi yang memegang peranan tersebut. Sequence diagram menangkap objek dan class yang terlibat dalam skenario dan urut-urutan pesan yang ditukar antara objek diperlukan untuk

melaksanakan fungsionalitas skenario. Sequence diagram berasosiasi dengan use case selama proses pengembangan.

1. Sequence Diagram Input Video

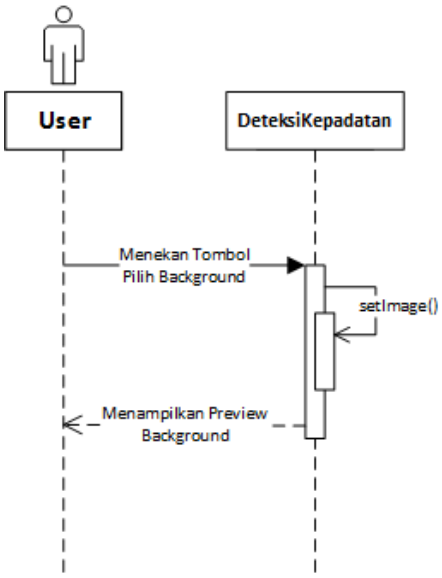
Sequence diagram input video menunjukkan rangkaian pesan yang dikirim antara objek User dengan objek lainnya dalam melakukkan proses input data rekaman video ke dalam sistem, sequence diagram input video dapat dilihat sebagai berikut:



Gambar 3.31 Sequence Diagram Input Video

2. Sequence Diagram Input Background

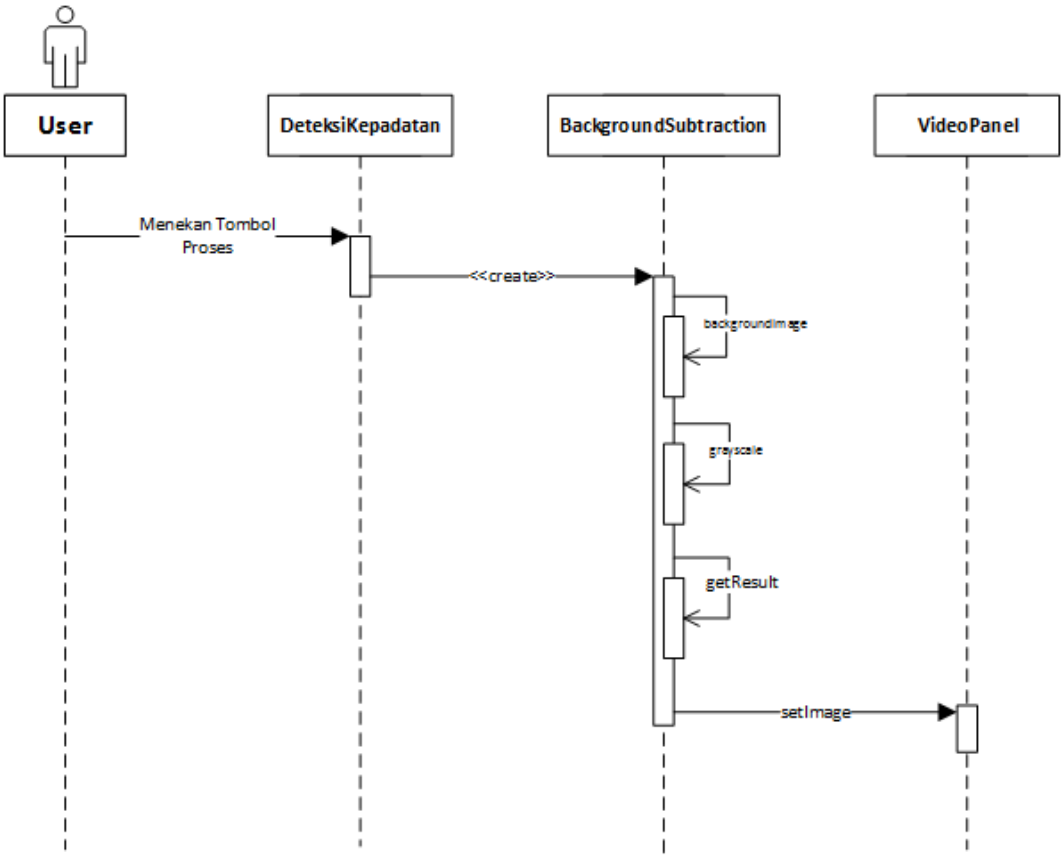
Sequence diagram input background menunjukkan rangkaian pesan yang dikirim antara objek User dengan objek lainnya dalam melakukkan proses input data citra background ke dalam sistem, sequence diagram input background dapat dilihat sebagai beirkut:



Gambar 3.32 Sequence Diagram Input Background

3. Sequence Diagram Proses Video

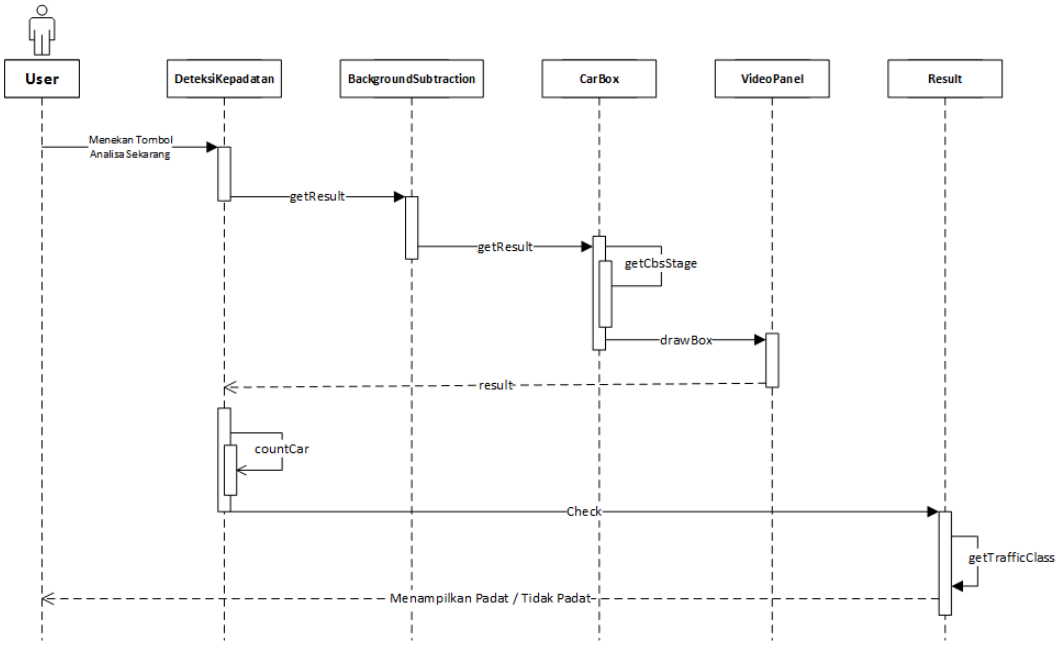
Sequence diagram proses video menunjukkan rangkaian pesan yang dikirim antara objek User dengan objek lainnya dalam melakukkan proses input data citra background ke dalam sistem, sequence diagram input background dapat dilihat sebagai beirkut:



Gambar 3.33 Sequence Diagram Proses Video

4. Sequence Diagram Penentuan Kepadatan

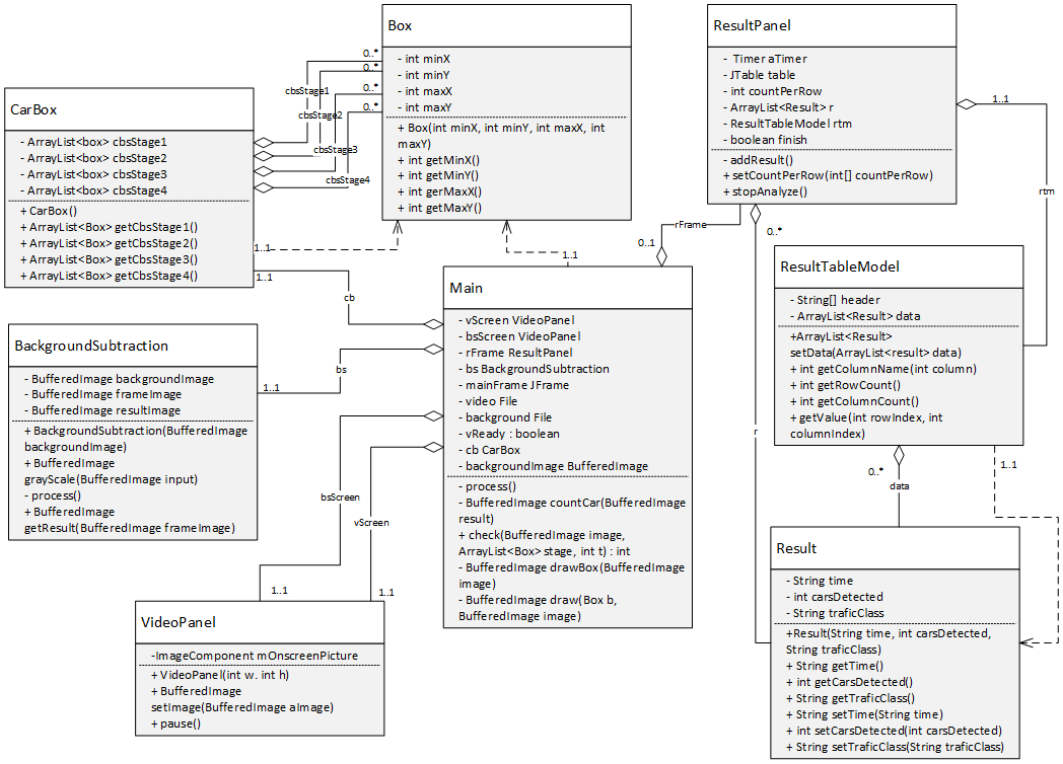
Sequence diagram analisis antrian menunjukkan rangkaian pesan yang dikirim antara objek User dengan objek lainnya dalam melakukkan proses input data citra background ke dalam sistem, sequence diagram input background dapat dilihat sebagai beirkut:



Gambar 3.34 Sequence Diagram Penentuan Kepadatan

3.5.3 Class Diagram

Class diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem. Struktur ini meliputi atribut-atribut dan metode-metode yang ada pada masing-masing kelas. Class diagram yang akan dibangun dapat dilihat pada gambar berikut :



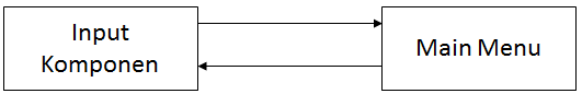
Gambar 3.35 Class Diagram

3.6 Perancangan Sistem

Pada sub bab ini akan membahas tentang perancangan sistem. Sub bab ini terdiri dari perancangan struktur menu, perancangan antar muka, perancangan pesan, perancangan jaringan semantik, dan perancangan method.

3.6.1 Perancangan Struktur Menu

Berikut ini adalah perancangan struktur menu dari aplikasi yang akan dibangun dapat dilihat pada gambar berikut :

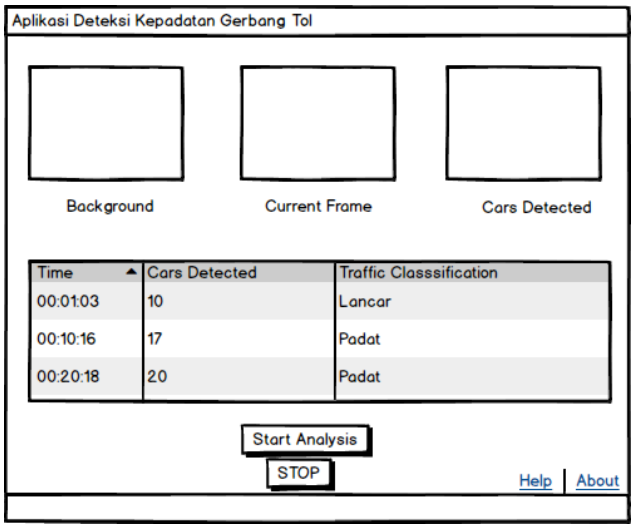


Gambar 3.36 Struktur Menu

3.6.2 Perancangan Antar Muka

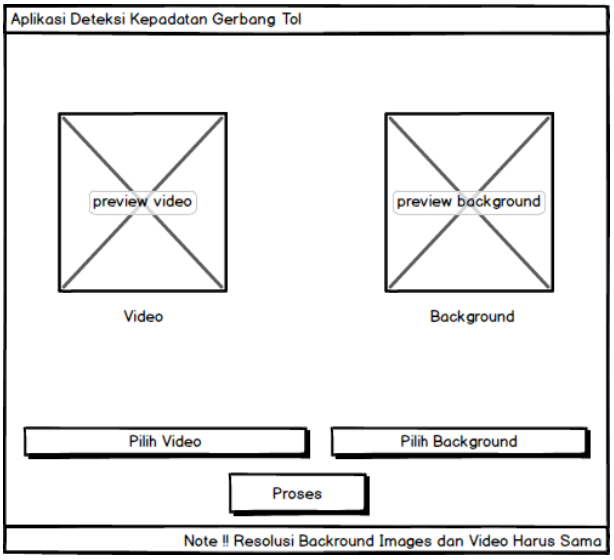
Perancangan antarmuka merupakan sebuah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Adapun perancangan antarmuka sistem yang akan dibuat adalah sebagai berikut :

1. Perancangan Antarmuka menu utama/main menu



Gambar 3.37 Antarmuka Main Menu

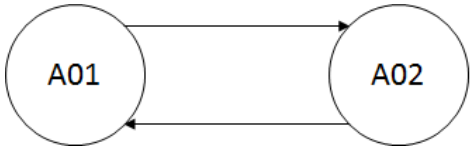
2. Perancangan antarmuka Input Komponen



Gambar 3.38 Perancangan Antar Muka Input Komponen

3.7 Jaringan Semantik

Perancangan jaringan semantik merupakan sebuah gambaran grafis yang berupa node-node yang menggambarkan aliran menu-menu yang ada di aplikasi yang akan dibangun. Perancangan jaringan semantik aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.39 Jaringan Semantik

Keterangan :

A01: Main Menu

A02: Input Component