BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem adalah tahapan sebelum melakukan perancangan program. Untuk memahami sistem, mengetahui keunggulan dan kelemahan dari perangkat lunak. Menganalisis tahapan membangun perangkat lunak *hybrid watermarking* dengan menggunakan DWT-DCT dan SVD dengan algoritma genetika. Perangkat lunak yang akan dibangun adalah aplikasi yang implementasikan teknik *watermarking* menggunakan DWT-DCT dan SVD dengan algoritma genetika sesuai kebutuhan-kebutuhan sebagai berikut:

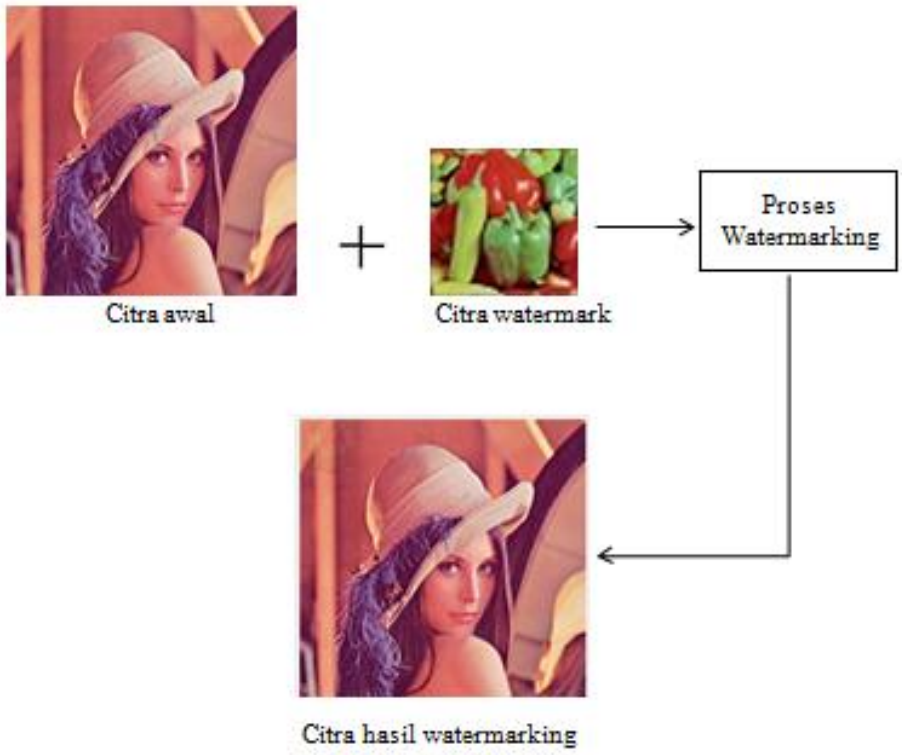
1. Mengimplementasikan teknik *watermarking* menggunakan metode DWT-DCT dan SVD dengan algoritma genetika sesuai dengan tahapan tahapannya.
2. Melakukan pengujian aplikasi untuk diukur performansi kerjanya dalam mengimplementasikan teknik *watermarking*.

3.1.1 Analisis Masalah

Analisis masalah dilakukan untuk mengetahui masalah apa saja yang terjadi pada pembangunan program. Masalah yang terjadi saat pembangunan program aplikasi *watermarking* pada citra digital yaitu kualitas citra yang disisipi *watermark*, apakah kulitasnya akan menurun atau menjadi lebih baik. Kemudian ukuran citra *watermark*, citra *watermark* yang akan disisipi harus lebih kecil dari ukuran citra awal. Bagaimana jika citra *watermark* yang disisipi ukurannya lebih besar dari citra awal. Kemudian format citra yang digunakan, format citra yang digunakan adalah berformat .jpg, .bmp dan .png. bagaimana jika format citra yang di inputkan bukan format tersebut. Kemudian ketahanan citra yang sudah diberi *watermark* terhadap serangan seperti rotasi dan *cropping*, apakah citra yang sudah diberi *watermark* akan tahan terhadap serangan.

3.1.2 Deskripsi Sistem

Deskripsi sistem merupan penjelasan secara umum atau gambaran secara umum dari sistem yang akan dibuat. Gambaran umum sistem dapat dilihat pada gambar 3.1.



Gambar 3.1 Gambaran Umum Sistem

Proses penyisipan *watermarking*

1. Masukkan citra awal
2. Masukkan citra *watermark*
3. Melakukan proses *watermarking*
4. Menampilkan citra hasil *watermarking*

Tahapan pertama adalah memasukkan citra awal, setelah memasukkan citra awal selanjutnya memasukkan citra *watermark*. Kemudian

melakukan proses *watermarking* dengan menggunakan teknik DWT-DCT dan SVD dengan optimasi algoritma genetika.

3.1.3 Analisis Citra Awal

Citra awal yang dapat diinputkan dapat berformat .jpg, .bmp, dan .png. Citra yang akan diubah kedalam bentuk matrik adalah citra Lena pada gambar 3.2 dengan ukuran 256 x 256 pixel.

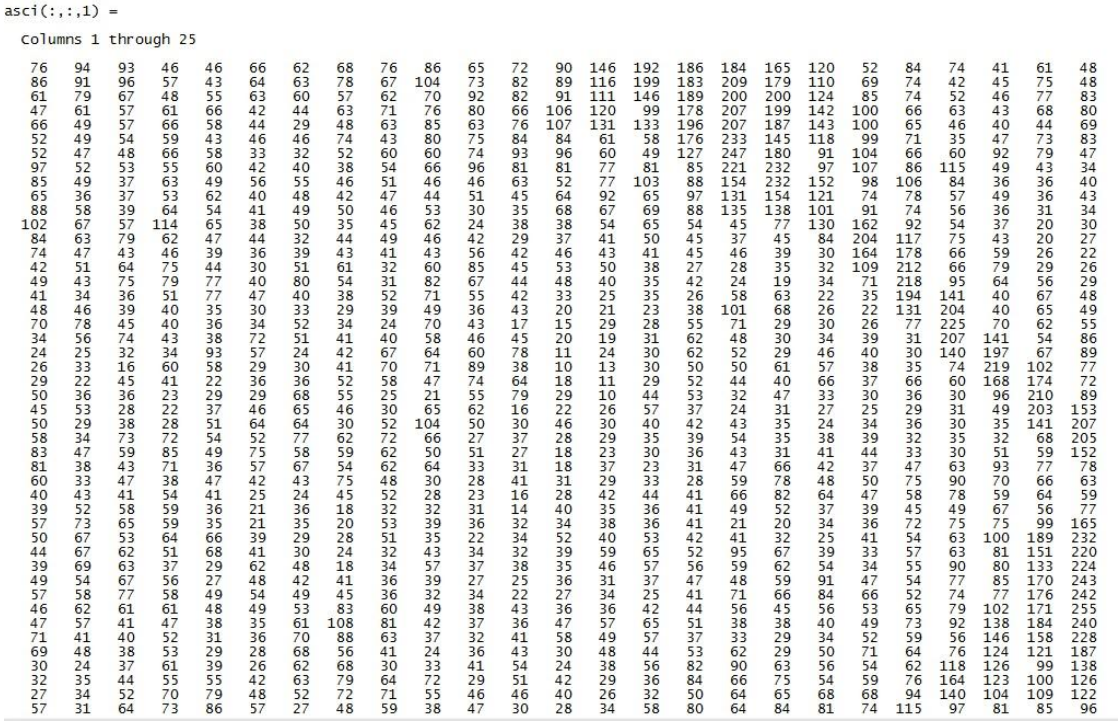


Gambar 3.2 Lena.jpg

Pada citra lena.jpg diatas ukurannya adalah 256 x 256 pixel yang artinya terdiri dari 256 kolom dan 256 baris pixel. Sehingga total keseluruhannya adalah 65536 pixel, setiap pixelnya perpaduan dari 3 warna yaitu *Red, Green* dan *Blue* atau sering disebut dengan RGB. Untuk mengubah kedalam bentuk matrik menggunakan aplikasi matlab, dengan mengetikkan perintah:

```
>>c=imread('lena.jpg','jpg');
>>asci=uint8(c)
```

Kemudian hasilnya seperti pada gambar dibawah ini



```
asci(:,:,1) =

  Columns 1 through 25

   76   94   93   46   46   66   62   68   76   86   65   72   90  146  192  186  184  165  120   52   84   74   41   61   48
   86   91   96   57   43   64   63   78   67  104   73   82   89  116  199  183  209  179  110   69   74   42   45   75   48
   61   79   67   48   55   63   60   57   62   70   92   82   91  111  146  189  200  200  124   85   74   52   46   77   83
   47   61   57   61   66   42   44   63   71   76   80   66  106  120   99  178  207  199  142  100   66   63   43   68   80
   66   49   57   66   58   44   29   48   63   85   63   76  107  131  133  196  207  187  143  100   65   46   40   44   69
   52   49   54   59   43   46   46   74   43   80   75   84   84   61   58  176  233  145  118   99   71   35   47   73   83
   52   47   48   66   58   33   32   52   60   60   74   93   96   60   49  127  247  180   91  104   66   60   92   79   47
   97   52   53   55   60   42   40   38   54   66   96   81   81   77   81   85  221  232   97  107   86  115   49   43   34
   85   49   37   63   49   56   55   46   51   46   46   63   52   77  103   88  154  232  152   98  106   84   36   36   40
   65   36   37   53   62   40   48   42   47   44   51   45   64   92   65   97  131  154  121   74   78   57   49   36   43
   88   58   39   64   54   41   49   50   46   53   30   35   68   67   69   88  135  138  101   91   74   56   36   31   34
  102   67   57  114   65   38   50   35   45   62   24   38   38   54   65   54   45   77  130  162   92   54   37   20   30
   84   63   79   62   47   44   32   44   49   46   42   29   37   41   50   45   37   45   84  204  117   75   43   20   27
   74   47   43   46   39   36   39   43   41   43   56   42   46   43   41   45   46   39   30  164  178   66   59   26   22
   42   51   64   75   44   30   51   61   32   60   85   45   53   50   38   27   28   35   32  109  212   66   79   29   26
   49   43   75   79   77   40   80   54   31   82   67   44   48   40   35   42   24   19   34   71  218   95   64   56   29
   41   34   36   51   77   47   40   38   52   71   55   42   33   25   35   26   58   63   22   35  194  141   40   67   48
   48   46   39   40   35   30   33   29   39   49   36   43   20   21   23   38  101   68   26   22  131  204   40   65   49
   70   78   45   40   36   34   52   34   24   70   43   17   15   29   28   55   71   29   30   26   77  225   70   62   55
   34   56   74   43   38   72   51   41   40   58   46   45   20   19   31   62   48   30   34   39   31  207  141   54   86
   24   25   32   34   93   57   24   42   67   64   60   78   11   24   30   62   52   29   46   40   30  140  197   67   89
   26   33   16   60   58   29   30   41   70   71   89   38   10   13   30   50   50   61   57   38   35   74  219  102   77
   29   22   45   41   22   36   36   52   58   47   74   64   18   11   29   52   44   40   66   37   66   60  168  174   72
   50   36   36   23   29   29   68   55   25   21   55   79   29   10   44   53   32   47   33   30   36   30   96  210   89
   45   53   28   22   37   46   65   46   30   65   62   16   22   26   57   37   24   31   27   25   29   31   49  203  153
   50   29   38   28   51   64   64   30   52  104   50   30   46   30   40   42   43   35   24   34   36   30   35  141  207
   58   34   73   72   54   52   77   62   72   66   27   37   28   29   35   39   54   35   38   39   32   35   32   68  205
   83   47   59   85   49   75   58   59   62   50   51   27   18   23   30   36   43   31   41   44   33   30   51   59  152
   81   38   43   71   36   57   67   54   62   64   33   31   18   37   23   31   47   66   42   37   47   63   93   77   78
   60   33   47   38   47   42   43   75   48   30   28   41   31   29   33   28   59   78   48   50   75   90   70   66   63
   40   43   41   54   41   25   24   45   52   28   23   16   28   42   44   41   66   82   64   47   58   78   59   64   59
   39   52   58   59   36   21   36   18   32   32   31   14   40   35   36   41   49   52   37   39   45   49   67   56   77
   57   73   65   59   35   21   35   20   53   39   36   32   34   38   36   41   21   20   34   36   72   75   75   99  165
   50   67   53   64   66   39   29   28   51   35   22   34   52   40   53   42   41   32   25   41   54   63  100  189  232
   44   67   62   51   68   41   30   24   32   43   34   32   39   59   65   52   95   67   39   33   57   63   81  151  220
   39   69   63   37   29   62   48   18   34   57   37   38   35   46   57   56   59   62   54   34   55   90   80  133  224
   49   54   67   56   27   48   42   41   36   39   27   25   36   31   37   47   48   59   91   47   54   77   85  170  243
   57   58   77   58   49   54   49   45   36   32   34   22   27   34   25   41   71   66   84   66   52   74   77  176  242
   46   62   61   61   48   49   53   83   60   49   38   43   36   36   42   44   56   45   56   53   65   79  102  171  255
   47   57   41   47   38   35   61  108   81   42   37   36   47   57   65   51   38   38   40   49   73   92  138  184  240
   71   41   40   52   31   36   70   88   63   37   32   41   58   49   57   37   33   29   34   52   59   56  146  158  228
   69   48   38   53   29   28   68   56   41   24   36   43   30   48   44   53   62   29   50   71   64   76  124  121  187
   30   24   37   61   39   26   62   68   30   33   41   54   24   38   56   82   90   63   56   54   62  118  126   99  138
   32   35   44   55   55   42   63   79   64   72   29   51   42   29   36   84   66   75   54   59   76  164  123  100  126
   27   34   52   70   79   48   52   72   71   55   46   46   40   26   32   50   64   65   68   68   94  140  104  109  122
   57   31   64   73   86   57   27   48   59   38   47   30   28   34   58   80   64   84   81   74  115   97   81   85   96
```

Gambar 3.3 Hasil Matriks

Pada gambar 3.3 terdapat asci(: , : , 1) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna pertama merah (*Red*).

```
asci(:,:,2) =

  Columns 1 through 24

   88  106  107   60   60   80   77   83   88   99   75   82   96  152  195  190  185  166  122   54   84   77   45   68
   98  106  110   73   57   78   77   93   80  117   85   92   97  122  205  187  213  181  113   72   77   46   52   82
   76   96   83   66   70   78   76   73   77   83  106   94  102  119  154  196  205  203  128   89   79   57   54   85
   65   81   76   79   84   60   62   79   88   92   97   80  119  131  110  185  212  205  149  108   73   69   51   76
   87   71   78   86   78   63   49   66   83  103   80   91  122  144  145  205  214  193  150  109   73   52   47   53
   75   71   77   81   62   65   65   94   62   97   92  102   99   75   70  188  242  153  129  110   79   44   56   81
   76   71   71   88   77   52   49   70   76   77   90  110  110   74   61  139  255  191  102  115   74   69  101   87
  119   75   74   75   79   59   57   52   69   80  111   94   94   91   93   97  232  244  110  121   98  126   58   51
  104   70   55   81   66   70   69   58   63   56   57   73   63   88  115   99  165  244  165  113  118   96   45   46
   80   51   52   68   75   52   57   51   56   52   59   54   74  103   76  108  143  168  135   90   92   71   60   46
  101   71   52   77   64   51   57   57   53   58   35   41   75   75   78   98  146  150  113  105   87   69   47   43
  112   77   67  124   73   46   56   40   49   66   28   43   44   61   73   62   54   88  142  176  106   67   50   31
   92   71   87   70   53   50   34   46   51   48   44   31   43   47   58   54   46   54   94  216  130   90   56   31
   82   55   51   54   45   42   41   45   43   45   58   47   51   50   48   52   53   48   40  178  193   81   72   37
   51   60   71   81   47   32   53   63   34   62   89   50   58   57   45   35   36   44   41  122  227   83   94   42
   58   52   82   85   79   42   81   58   35   87   73   50   54   46   41   48   32   30   45   85  234  112   79   70
   51   44   44   57   79   48   44   44   60   79   63   49   37   29   36   30   66   73   34   51  213  161   56   80
   58   56   47   47   40   34   38   36   50   60   47   52   24   24   24   40  107   78   38   35  146  221   57   79
   82   87   52   47   40   38   57   41   35   83   55   28   20   32   32   60   76   36   38   35   90  240   85   73
   45   67   80   49   42   78   58   51   53   74   62   59   28   24   37   67   54   35   40   46   41  218  153   64
   33   34   37   40   99   63   32   53   84   83   78   93   21   32   35   69   59   36   52   46   37  149  207   76
   34   38   20   65   64   37   38   54   87   91  107   55   20   21   37   56   59   70   65   45   42   82  228  112
   34   27   49   46   28   44   45   66   75   67   93   81   29   19   36   60   56   52   77   46   73   68  178  185
   56   42   40   28   35   35   76   68   42   40   73   94   40   19   49   59   43   59   44   38   43   38  106  221
   52   59   32   26   41   52   72   56   43   81   77   31   31   32   61   41   32   39   34   30   35   37   58  214
   58   37   43   32   55   70   70   40   64  119   65   42   52   35   41   44   47   40   27   37   36   32   44  152
   69   44   80   78   58   58   83   72   84   78   39   48   34   32   34   39   53   33   34   35   28   35   37   78
   94   57   68   92   56   82   66   69   74   62   63   37   24   25   29   33   40   28   36   39   28   28   54   66
   94   51   54   79   45   66   79   67   75   79   45   41   22   36   18   26   45   65   39   34   44   63   96   83
   75   47   60   49   59   54   57   91   64   45   40   51   35   28   28   25   60   80   47   50   75   90   73   71
   58   60   57   67   54   38   37   61   67   45   38   28   32   42   40   38   71   88   67   50   60   81   62   69
   57   70   72   72   49   34   52   33   47   47   43   24   45   35   33   41   56   61   44   45   47   48   65   52
   69   84   78   71   48   34   47   32   65   49   43   37   37   39   34   42   29   30   44   43   72   69   60   78
   60   77   63   76   78   52   41   40   61   44   28   39   55   42   54   45   48   40   33   46   52   52   76  156
   54   76   72   63   80   54   45   39   44   53   44   39   43   61   68   55   98   73   44   34   51   46   52  112
   48   78   71   46   41   75   64   34   47   69   47   48   41   50   63   59   61   64   55   33   47   70   49   93
   55   62   74   65   39   61   58   57   52   54   39   35   44   38   43   50   48   59   90   45   46   58   55  128
   59   63   82   65   60   67   63   62   52   48   47   34   35   42   32   44   71   66   83   64   46   60   49  139
   46   63   65   67   56   61   67   97   77   65   51   56   44   44   49   50   57   46   59   56   64   70   82  140
   46   56   42   51   44   44   74  123   95   56   50   47   55   65   72   57   40   40   46   54   77   88  124  159
   67   39   39   53   37   45   82  101   76   50   44   52   66   56   62   43   35   31   40   59   65   56  136  139
   64   44   36   53   33   37   78   69   54   37   48   52   37   55   50   56   65   32   54   76   70   77  118  108
   26   20   35   62   43   35   72   81   43   46   50   63   31   43   59   86   92   66   59   57   65  118  120   87
   28   34   44   58   61   51   75   92   77   85   38   60   49   35   39   89   70   76   58   62   77  164  119   91
   26   35   55   76   87   59   64   86   86   69   58   55   48   32   36   55   69   70   70   98  143  105  105
   58   35   70   81   97   70   41   62   74   52   59   42   38   43   66   86   71   90   86   79  120  103   88   88
  103   79   84   54   84   89   57   59   61   76   74   66   61  100   97   83   61   98   82   89  114  104   89   90
```

Gambar 3.4hasil matriks

Pada gambar 3.4 terdapat asci(: , : , 2) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna kedua yaitu hijau (*Green*).

```
asci(:,:,3) =

  Columns 1 through 24

   26   44   45    0    0   18   12   16   22   29    6   11   24   80  124  114  105   86   47    0   34   30    0   16
   34   41   48   10    0   16   15   28   11   47   13   19   24   50  133  111  129   97   34    3   22    0    0   28
    7   28   18    4   11   19   14    8    8   11   31   18   26   44   81  118  115  112   43   12   15    1    0   28
    0    9   10   17   26    2    2   16   18   19   19    1   40   53   34  107  120  109   56   23    5    9    0   19
   10    0    9   23   19    7    0    4   14   29    0   10   41   65   69  126  121   95   54   20    0    0    0    0
    0    0    7   17    6   10    9   33    0   26   14   20   18    0    0  106  149   52   27   18    4    0    3   32
    0    0    1   24   21    0    0   12   13    9   17   32   33    0    0   57  163   89    0   21    0    6   48   40
   47    3    9   14   23    5    3    0   12   19   44   24   22   16   19   15  138  144    5   23   16   58    5    4
   40    5    0   21   11   17   18    8   13    3    1   12    0   19   43   20   73  144   60   12   32   24    0    0
   21    0    0   11   21    2   10    4   11    5   10    1   14   37    7   32   57   73   37    0    5    0    2    0
   47   17    0   23   12    1   10   13   11   18    0    0   23   16   15   29   70   68   27   18    5    0    0    0
   60   25   15   72   22    0   12    0   12   29    0    2    0    9   16    2    0   19   68   97   29    0    0    0
   41   20   36   21    5    4    0    7   14   11    7    0    0    0    7    1    0    0   33  150   60   23    3    0
   31    4    2    5    0    0    2    8    6    8   21    7   10    6    4    8   11    1    0  117  126   16   19    0
    0    7   20   35    4    0   14   26    0   25   52   10   18   16    4    0    0    1    0   66  160   15   39    0
    5    0   31   39   39    5   47   23    0   47   35   12   20   12    5   10    0    0    2   32  169   44   20   21
    0    0    0   11   40   14   11   10   19   38   24   16   14    6    5    0   29   36    0    4  149   90    0   24
    8    6    0    3    0    0    6    2    8   17    7   21    7    5    0    3   69   41    2    0   87  153    0   18
   34   40    8    5    3    5   27    8    0   39   15    0    0   11    0   20   35    0    1    0   34  175   20   13
    5   25   42   11    7   44   25   14    9   25   17   23    4    1    3   26   10    0    2    5    0  158   89    3
    4    3    5    6   65   27    0   10   32   28   28   52    0    8    3   28   15    0   14    8    0   96  144   11
   10   15    0   35   30    0    0    8   32   32   55   11    0    0    4   18   14   25   24    4    0   31  163   41
   12    4   24   16    0    5    2   17   20    8   38   36    0    0    5   21   10    4   34    3   31   17  109  107
   28   14   13    0    1    0   35   22    0    0   23   51   10    0   17   21    0   13    2    0    2    0   37  142
   11   21    0    0    8   16   38   19    0   36   34    0    2    4   26    4    0    2    1    0    1    0    0  135
    9    0    3    0   20   36   36    5   26   78   26    6   24    5    7    7   14   10    0    8    8    0    0   76
   13    0   38   40   21   22   49   37   48   40    3   14    6    5    3    5   23    8    9    8    1    0    0    7
   38    4   23   51   15   41   27   32   36   24   25    3    0    1    1    2    9    0    6    9    0    0    0    0
   40    0   11   38    2   21   33   23   32   36    5    6    0   15    0    0    6   19    0    0    0   13   39   19
   18    0   16    7   13    6    8   44   19    2    0   16   10    7    6    0   16   30    0    0   25   36   16    7
    0    5   10   24    8    0    0   16   24    3    0    0    5   18   13    7   30   42   22    3   10   28    7    5
    0   12   23   28    3    0    7    0    6    8    7    0   15    7    2    5   15   18    2    1    0    0    6    0
    7   24   25   23    2    0    7    0   29   15   12    7    6    5    0    2    0    0    7    1   20    7    0    0
    0   17   10   26   32    9    1    2   27   13    0    9   24    5   12    0    6    1    0    6    1    0    4   61
    0   19   20   15   34   10    2    0    6   18   10    6   10   24   25    8   51   27    3    0    3    0    0   19
    0   25   22    0    0   31   17    0    4   31   12   13    5   13   19   12   11   14   11    0    0   17    0    0
    7   13   30   20    0   15    9    8    5   11    0    0    5    0    0    3    0    7   42    0    0    2    0   30
   19   22   42   24   17   21   12   10    3    1    4    0    0    1    0    0   21   16   35   15    0    0    0   32
   12   29   30   29   15   15   16   44   23   16    7   13    5    3    7    6   15    2   12    3    7    1    0   24
   16   26   11   18    6    0   21   68   42    5    6    5   16   24   30   13    1    0    0    0   16   15   35   40
   42   14   11   22    1    2   32   47   22    0    0    9   25   15   21    0    0    0    0    5    5    0   47   19
   42   19   11   25    0    0   28   15    0    0    2    9    0   13    6   11   22    0    4   22   10    7   32    0
    1    0   10   31    8    0   22   28    0    0    3   18    0    2   14   36   42   13    6    2    8   54   42    0
    1    6   16   27   25    8   25   39   23   32    0   15    5    0    0   35   11   16    0    5   20  102   45    0
    0    4   24   42   48   16   14   33   31   16    8    8    1    0    0    0    3    3    5    7   37   76   29   14
   24    0   34   44   55   24    0    9   19    0    9    0    0    0    9   24    1   18   19   12   53   29    8    0
```

Gambar 3.5 hasil matriks

Pada gambar 3.5 terdapat asci(: , : , 3) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna ketiga yaitu biru (*Blue*).

3.1.4 Analisis Citra *watermark*

Citra *watermark* yang dapat diinputkan dapat berformat .jpg, .bmp, dan .png. Ukuran citra *watermark* harus lebih kecil dari ukuran citra awal. Selanjutnya citra *watermark* yang akan diubah kedalam bentuk matrik adalah citra paprika pada gambar 3.6 dengan ukuran 50 x 50 pixel.



Gambar 3.6 paprika.jpg

Pada citra paprika.jpg diatas ukurannya adalah 50 x 50 pixel yang artinya terdiri dari 50 kolom dan 50 baris pixel. Sehingga total keseluruhannya adalah 2500 pixel, setiap pixelnya perpaduan dari 3 warna yaitu Red, Green dan Blue atau sering disebut dengan RGB. Untuk mengubah kedalam bentuk matrik disini saya menggunakan aplikasi matlab, dengan mengetikkan perintah:

>>c=imread(‘paprika.jpg’,’jpg’);
>>asci=uint8(c)

Kemudian hasilnya seperti pada gambar dibawah ini

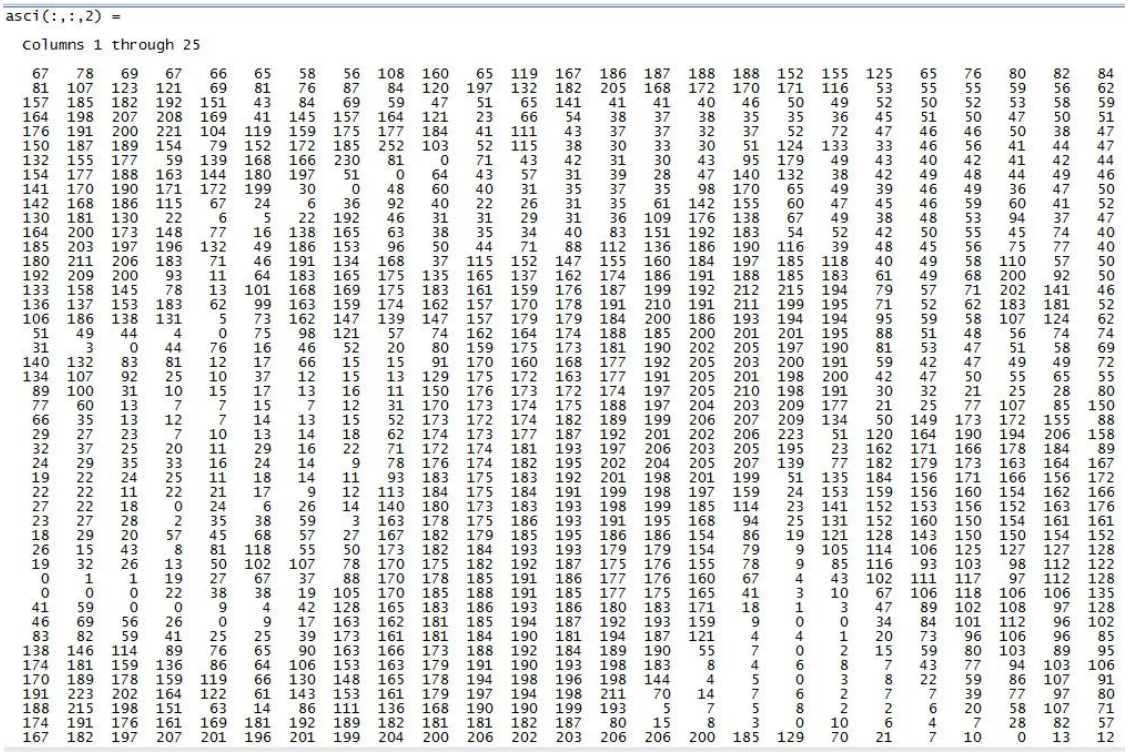
```
asci(:,:,1) =

  Columns 1 through 25

  140  152  146  150  165  171  168  149  167  191   80  121  164  181  186  182  172  147  185  191  166  192  195  184  162
  140  166  180  185  154  174  167  163  138  158  228  156  205  233  206  210  198  207  177  138  165  176  180  168  159
  190  211  198  217  212  111  135  111  106   99  112  135  220  135  157  165  162  165  168  175  177  183  188  194  191
  179  207  207  220  212   87  163  167  185  158   83  151  164  174  194  202  189  179  174  182  187  187  189  197  201
  180  197  211  242  138  142  153  153  157  179   69  181  164  189  202  191  174  174  192  171  183  188  196  186  196
  136  180  197  169   91  150  149  153  227  103   90  199  170  189  198  178  157  219  246  161  187  203  188  191  194
   94  127  166   53  119  139  134  210   94   47  164  169  186  186  189  170  158  237  167  191  187  189  183  184  187
  106  137  163  139  112  149  172   53   42  154  175  209  184  188  168  151  184  181  158  196  201  197  188  193  189
   98  126  150  138  151  190   35   30  137  185  190  194  196  179  143  176  222  131  169  186  198  202  191  200  197
  103  129  151   92   65   34   20   80  198  184  183  191  192  165  135  190  211  141  170  192  198  214  218  194  199
   94  153  116   26   33   32   25  221  152  181  191  190  185  148  146  193  193  161  180  190  203  208  245  184  192
  136  178  166  162  116   47  131  178  156  171  172  163  151  151  153  183  217  134  178  191  203  206  187  214  183
  167  187  183  198  165   75  170  152  164  149  133  139  122  114  109  155  181  151  146  192  193  199  208  208  180
  165  196  194  187  102   69  173  120  203   92  161  177  140  128  122  145  168  195  198  165  191  196  221  166  182
  170  195  200  110   45   87  164  138  172  143  166  132  148  153  159  160  164  184  232  153  182  197  255  169  168
  102  135  137   87   42  120  148  137  153  165  137  136  156  166  174  166  188  208  222  152  187  198  255  207  158
   96   98  117  162   74  111  142  125  149  138  127  140  154  169  187  166  185  188  217  141  183  195  255  254  168
   76  156  105  113   19   92  162  139  139  138  128  143  151  159  180  163  167  182  214  160  185  197  209  220  183
   55   52   48   19   28  121  143  168  109  104  144  126  141  161  166  181  176  189  215  152  170  191  192  206  200
   51   28   29   93  138   89  128  136   97  126  148  137  140  153  168  182  182  187  211  141  167  190  203  208  201
  158  166  149  170  110  121  174  116   95  134  158  123  139  150  165  179  183  195  213  119  151  185  199  199  213
  169  159  183  138  129  158  136  124   93  167  163  137  134  151  163  178  181  192  218   95  144  173  185  191  163
  160  184  139  132  137  141  140  126   86  184  165  142  143  148  174  183  187  186  204   72  107  113  120  105  118
  175  164  132  135  136  144  137  120   96  195  162  145  145  161  176  184  180  195  191   51   64  120  150  113  151
  180  153  139  145  141  148  142  118  104  188  160  148  154  163  181  188  185  198  150   67  142  151  144  132   85
  151  151  153  140  145  147  144  118  109  182  159  150  158  165  181  183  185  211   60  122  128  136  136  162  146
  158  164  155  153  144  162  148  125  117  179  155  150  163  168  178  180  185  180   13  143  127  111  132  145   61
  150  155  167  167  149  156  144  110  121  179  154  153  166  174  179  182  195  126   55  146  128  119  120  123  124
  140  146  158  164  147  151  139  103  127  180  155  156  170  182  180  189  197   44  104  139  104  117  116  104  113
  140  145  143  155  150  136  117   86  136  175  155  161  174  184  181  187  164   20  123  112  109  110  102  104  103
  148  143  139  115  129   96  101   59  145  165  158  167  175  178  177  170  120   26  115  113  117  117  103  107  122
  141  141  134   96  114  103  111   32  160  159  160  169  175  169  172  154  107   38  114  124  132  119  109  111  114
  134  135  105  122   94  110   95   50  163  162  158  164  173  165  164  148  110   45  120  111  122  121  111  111  106
  137  113  120   64  118  148   89   71  166  161  161  171  174  161  160  148  108   41  114  108   93  108  106   98   95
  122  128  109   79   94  137  142   99  163  155  163  175  172  160  156  148  101   40  103  125   95  105  100  109  108
   85   90   87   93   81  108   73  108  161  156  166  175  173  162  156  153   92   42   72  122  123  125  111  120  123
   71   63   85  107  105   90   58  121  157  158  166  173  168  160  155  163   76   53   53   99  120  125  118  111  123
  102  126   85   83   83   61   77  139  150  156  166  174  167  159  159  168   53   56   53   86  105  111  125  107  120
  100  129  131  104   68   64   43  165  147  159  169  178  166  167  166  149   39   48   48   76  108  119  136  116  107
  123  129  120  110  100   82   61  170  147  163  171  174  160  170  167  120   36   55   56   69  104  119  134  118   95
  161  175  156  147  153  129  114  162  155  158  171  174  162  174  189   73   49   55   63   70   97  107  127  106   99
  180  191  180  175  153  124  124  149  154  164  171  167  171  187  193   41   58   71   77   71   92  114  120  117  108
  163  182  178  174  165  108  136  135  151  161  171  172  170  186  159   44   69   75   80   83   85  110  120  125   94
  179  212  192  170  156   93  146  143  150  164  176  170  177  206   93   60   76   86   84   86   78   98  122  126   94
  182  209  194  159   94   45   97  111  133  161  177  177  191  199   39   61   74   84   78   78   79   86  115  152  103
  173  190  177  168  187  202  201  192  181  175  172  173  183   86   43   49   54   57   67   65   66   69   87  137  102
  174  188  201  208  202  193  200  195  197  192  199  193  195  199  212  211  201  150   93   50   44   55   56   72   69
```

Gambar 3.7 Hasil Matriks

Pada gambar 3.7 terdapat asci(: , : , 1) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna pertama merah (Red).



```
asci(:,:,2) =

  Columns 1 through 25

   67   78   69   67   66   65   58   56  108  160   65  119  167  186  187  188  188  152  155  125   65   76   80   82   84
   81  107  123  121   69   81   76   87   84  120  197  132  182  205  168  172  170  171  116   53   55   55   59   56   62
  157  185  182  192  151   43   84   69   59   47   51   65  141   41   41   40   46   50   49   52   50   52   53   58   59
  164  198  207  208  169   41  145  157  164  121   23   66   54   38   37   38   35   35   36   45   51   50   47   50   51
  176  191  200  221  104  119  159  175  177  184   41  111   43   37   37   32   37   52   72   47   46   46   50   38   47
  150  187  189  154   79  152  172  185  252  103   52  115   38   30   33   30   51  124  133   33   46   56   41   44   47
  132  155  177   59  139  168  166  230   81    0   71   43   42   31   30   43   95  179   49   43   40   42   41   42   44
  154  177  188  163  144  180  197   51    0   64   43   57   31   39   28   47  140  132   38   42   49   48   44   49   46
  141  170  190  171  172  199   30    0   48   60   40   31   35   37   35   98  170   65   49   39   46   49   36   47   50
  142  168  186  115   67   24    6   36   92   40   22   26   31   35   61  142  155   60   47   45   46   59   60   41   52
  130  181  130   22    6    5   22  192   46   31   31   29   31   36  109  176  138   67   49   38   48   53   94   37   47
  164  200  173  148   77   16  138  165   63   38   35   34   40   83  151  192  183   54   52   42   50   55   45   74   40
  185  203  197  196  132   49  186  153   96   50   44   71   88  112  136  186  190  116   39   48   45   56   75   77   40
  180  211  206  183   71   46  191  134  168   37  115  152  147  155  160  184  197  185  118   40   49   58  110   57   50
  192  209  200   93   11   64  183  165  175  135  165  137  162  174  186  191  188  185  183   61   49   68  200   92   50
  133  158  145   78   13  101  168  169  175  183  161  159  176  187  199  192  212  215  194   79   57   71  202  141   46
  136  137  153  183   62   99  163  159  174  162  157  170  178  191  210  191  211  199  195   71   52   62  183  181   52
  106  186  138  131    5   73  162  147  139  147  157  179  179  184  200  186  193  194  194   95   59   58  107  124   62
   51   49   44    4    0   75   98  121   57   74  162  164  174  188  185  200  201  201  195   88   51   48   56   74   74
   31    3    0   44   76   16   46   52   20   80  159  175  173  181  190  202  205  197  190   81   53   47   51   58   69
  140  132   83   81   12   17   66   15   15   91  170  160  168  177  192  205  203  200  191   59   42   47   49   49   72
  134  107   92   25   10   37   12   15   13  129  175  172  163  177  191  205  201  198  200   42   47   50   55   65   55
   89  100   31   10   15   17   13   16   11  150  176  173  172  174  197  205  210  198  191   30   32   21   25   28   80
   77   60   13    7    7   15    7   12   31  170  173  174  175  188  197  204  203  209  177   21   25   77  107   85  150
   66   35   13   12    7   14   13   15   52  173  172  174  182  189  199  206  207  209  134   50  149  173  172  155   88
   29   27   23    7   10   13   14   18   62  174  173  177  187  192  201  202  206  223   51  120  164  190  194  206  158
   32   37   25   20   11   29   16   22   71  172  174  181  193  197  206  203  205  195   23  162  171  166  178  184   89
   24   29   35   33   16   24   14    9   78  176  174  182  195  202  204  205  207  139   77  182  179  173  163  164  167
   19   22   24   25   11   18   14   11   93  183  175  183  192  201  198  201  199   51  135  184  156  171  166  156  172
   22   22   11   22   21   17    9   12  113  184  175  184  191  199  198  197  159   24  153  159  156  160  154  162  166
   27   22   18    0   24    6   26   14  140  180  173  183  193  198  199  185  114   23  141  152  153  156  152  163  176
   23   27   28    2   35   38   59    3  163  178  175  186  193  191  195  168   94   25  131  152  160  150  154  161  161
   18   29   20   57   45   68   57   27  167  182  179  185  195  186  186  154   86   19  121  128  143  150  150  154  152
   26   15   43    8   81  118   55   50  173  182  184  193  193  179  179  154   79    9  105  114  106  125  127  127  128
   19   32   26   13   50  102  107   78  170  175  182  192  187  175  176  155   78    9   85  116   93  103   98  112  122
    0    1    1   19   27   67   37   88  170  178  185  191  186  177  176  160   67    4   43  102  111  117   97  112  128
    0    0    0   22   38   38   19  105  170  185  188  191  185  177  175  165   41    3   10   67  106  118  106  106  135
   41   59    0    0    9    4   42  128  165  183  186  193  186  180  183  171   18    1    3   47   89  102  108   97  128
   46   69   56   26    0    9   17  163  162  181  185  194  187  192  193  159    9    0    0   34   84  101  112   96  102
   83   82   59   41   25   25   39  173  161  181  184  190  181  194  187  121    4    4    1   20   73   96  106   96   85
  138  146  114   89   76   65   90  163  166  173  188  192  184  189  190   55    7    0    2   15   59   80  103   89   95
  174  181  159  136   86   64  106  153  163  179  191  190  193  198  183    8    4    6    8    7   43   77   94  103  106
  170  189  178  159  119   66  130  148  165  178  194  198  196  198  144    4    5    0    3    8   22   59   86  107   91
  191  223  202  164  122   61  143  153  161  179  197  194  198  211   70   14    7    6    2    7    7   39   77   97   80
  188  215  198  151   63   14   86  111  136  168  190  190  199  193    5    7    5    8    2    2    6   20   58  107   71
  174  191  176  161  169  181  192  189  182  181  181  182  187   80   15    8    3    0   10    6    4    7   28   82   57
  167  182  197  207  201  196  201  199  204  200  206  202  203  206  206  200  185  129   70   21    7   10    0   13   12
```

Gambar 3.8 Hasil Matriks

Pada gambar 3.8 terdapat asci(: , : , 2) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna kedua yaitu hijau (Green).

```
asci(:,:,3) =

  Columns 1 through 25

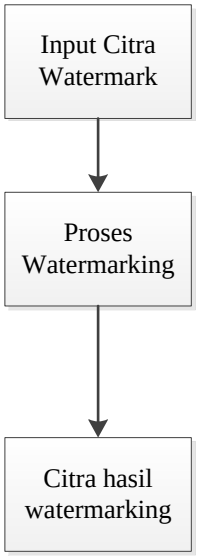
   24   29   13   15   35   43   35   25   64  106   10   60  110  132  147  144  126   86   95   77   35   53   61   57   48
   25   47   56   59   30   50   45   51   40   73  151   84  132  158  132  135  122  119   71   16   28   34   38   32   30
   76  101   94  110   94    0   37   21   15    8   23   39  108    7   16   20   23   29   29   34   31   32   31   36   38
   69  103  111  122  100    0   79   95  111   79    0   46   29   16   22   28   25   24   26   35   37   32   27   32   37
   77   95  111  140   30   41   73   90  106  128    2   85   24   26   35   28   21   31   56   35   36   34   35   24   40
   53   94  104   71    0   68   82  100  187   53   15   89   23   24   37   26   27   96  115   22   37   48   31   34   40
   45   68   85    0   50   84   90  169   46    0   53   28   31   26   34   36   60  141   35   39   33   35   31   32   36
   70   88   94   69   59  110  142   13    0   56   38   52   26   35   27   34  101   92   24   40   44   42   36   40   38
   52   75   91   80  107  156    8    0   40   56   41   36   41   36   25   75  133   30   33   32   43   44   32   42   42
   53   77   92   35   20    0    0   25   79   32   27   33   39   33   36  104  118   30   32   38   41   55   57   36   45
   43   97   51    0    0    0    0  158   20   16   31   34   33   22   64  122   99   42   39   35   43   48   87   29   42
   79  117   93   87   46    0   87  112   29   19   29   29   23   48   90  127  137   27   40   36   44   44   33   61   32
   99  114   99  111   79    0  115   85   59   31   30   48   43   51   65  116  133   76   21   39   35   40   58   59   27
   95  120  106   94   16    0  115   57  114    0   82  111   79   74   75  105  130  132   85   20   35   45   91   37   37
  109  124  114   23    0   30  119   84   88   54  100   71   77   79   89   98  110  119  127   22   34   62  186   76   40
   53   78   62   13    0   71  107   84   77   84   77   77   81   86   98   95  124  135  121   28   41   65  190  125   35
   63   58   66  108   20   59   94   72   82   68   69   80   80   91  119   99  114  103  110   12   32   57  170  166   39
   46  124   69   71    0   33  100   72   67   68   67   81   77   83  111   96   92   94  105   31   34   51   93  108   45
   14   16   17    0    0   51   57   75   17   22   78   63   69   85   93  108  100  101  109   26   21   34   42   59   52
    0    0    0   39   65    1   24   28    0   44   81   76   69   79   92  103   99   98  107   21   26   33   37   41   46
   78   84   61   73    1    4   56    5    4   59   98   65   75   80   89   96   90   98  115    9   21   37   35   32   52
   76   60   71   17    2   29   12   12    2   93  103   80   73   80   81   88   80   92  126    0   30   42   39   43   29
   61   76   19    5   12   17   20   15    0  104   98   80   79   73   81   79   80   86  121    0   13    8    3    0   41
   68   51    9    6   11   20   15    9    1  116   95   82   79   83   80   79   71   98  114    0    0   45   72   45  104
   65   33   14   17   16   23   18    6   15  114   96   85   82   82   87   88   81  104   75    7   95  111  111   99   45
   26   25   25   12   17   20   16    3   20  111   94   84   81   79   88   86   87  123    0   72   94  112  120  143  110
   20   28   23   21   14   30   14    5   22  100   85   77   79   78   85   86   94  102    0  106   94   82  105  117   30
    9   17   31   34   17   22   12    0   25   97   79   74   79   83   84   91  109   57   13  118  100   89   91   94   96
    8   14   21   28   13   19   12    0   30   96   76   78   84   93   88  101  116    0   68  117   81   93   93   82   92
   12   14    7   23   25   19    7    0   46   93   76   80   87   94   92  101   91    0   89   91   86   89   82   86   85
   16   13   10    0   28    8   21    0   72   89   78   84   85   87   88   90   56    0   80   85   89   93   86   92  101
   11   16   15    0   31   34   48    0   92   86   80   84   85   80   87   81   49    0   77   91  101   91   89   92   89
    5   15    0   37   28   52   38    0   90   83   76   80   87   81   88   80   50    0   77   74   87   92   87   85   77
   19    4   25    0   55   92   28    5   93   81   78   85   86   79   87   84   45    0   66   66   53   71   70   63   57
   23   33   18    0   23   72   77   31   90   76   77   86   82   74   81   78   37    0   47   73   45   55   49   59   61
    2    5    0    8    3   37    5   38   89   78   78   84   81   76   79   82   27    0   13   67   69   71   52   63   70
    0    0    0   19   22   14    0   54   88   82   77   79   79   75   76   90   11    0    0   42   69   72   58   51   73
   23   43    0    0    0    0   10   74   82   80   73   78   79   79   85  100    0    0    0   30   56   59   62   44   68
    8   35   37   13    0    0    0  106   79   80   78   85   82   91  100   90    0    0    0   22   56   63   68   46   47
   22   28   15    8    2    0    0  106   74   81   78   83   76   96  102   61    0    0    0   13   52   62   66   49   34
   58   68   40   26   34   21   28   85   74   72   82   84   76   94  120    9    0    0    1   10   40   51   65   45   47
   90   94   70   57   31   10   34   68   70   76   80   76   82  102  121    0    0    4   11    5   29   51   59   58   57
  102  112   90   76   60    8   54   60   70   72   77   73   73   98   87    0    5    9   13   13   15   40   51   61   38
  129  154  115   80   59    0   64   64   67   76   80   72   79  119   26    1   10   17   14   12    3   23   48   57   31
  126  145  113   66    0    0    7   25   49   73   85   84  100  119    0    5    8   12    4    2    0    6   38   76   30
  104  115   86   70   87  100   99   92   88   85   76   77   90    4    0    0    0    0    0    0    0    0   12   62   28
   86   94   97   99   93   83   81   78   90   89   92   87   91  103  128  134  123   72   16    0    0    0    0    7    1
```

Gambar 3.9 Hasil Matriks

Pada gambar 3.9 terdapat asci(: , : , 3) menunjukkan bahwa pada matriks yang ditampilkan adalah komposisi matriks dari warna ketiga yaitu biru (Blue).

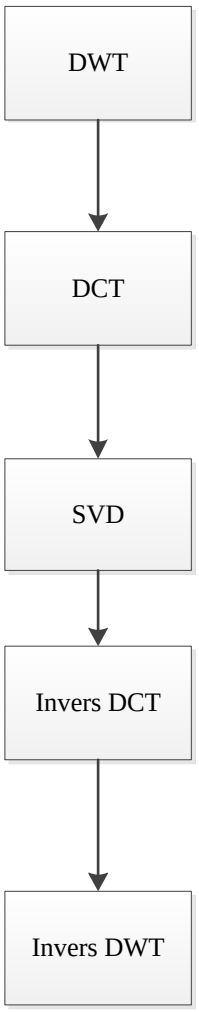
3.1.5 Analisis proses *watermarking*

Proses *watermarking* dapat dilihat pada gambar 3.10.



Gambar 3.10 Alur Proses *Watermarking* Pada Citra Digital Menggunakan Teknik DWT-DCT-SVD Dengan Algoritma Genetika

Gambar 3.10 menunjukkan alur proses *watermarking* dengan hybrid *watermarking* DWT-DCT-SVD dengan algoritma genetika. Tahapannya adalah input citra digital kemudian input citra *watermark* setelah itu melakukan proses *watermarking* dan menghasilkan citra yang ber-watermark.

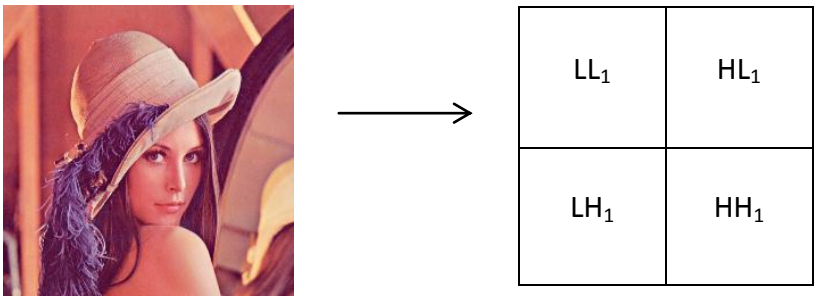


Gambar 3.11 Alur Hybrid *Watermarking* Menggunakan Teknik DWT-DCT-SVD dengan Algoritma Genetika

Gambar 3.11 menunjukkan alur hybrid *watermarking* dengan menggunakan teknik DWT-DCT-SVD dengan algoritma genetika. Tahapan pertama yang dilakukan adalah teknik DWT dengan optimasi algoritma genetika untuk memilih bagian yang disisipi *watermark*, kemudian melakukan teknik DCT setelah itu teknik SVD selanjutnya melakukan invers DCT dan tahapan terakhir adalah invers DWT.

Algoritma penyisipan watermark

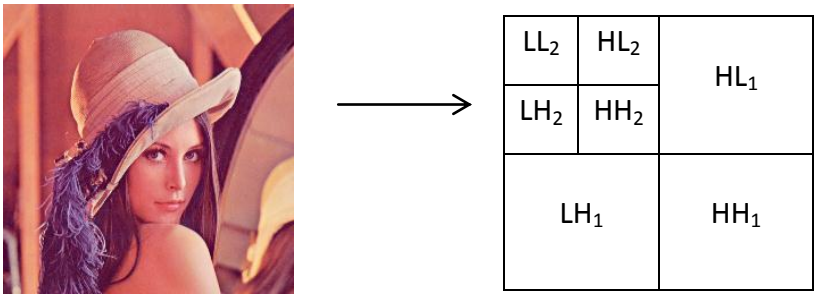
1. Terapkan DWT pada citra untuk mendekomposisi citra menjadi empat bagian frekuensi yaitu LL, HL, LH, dan HH.



Gambar 3.12 Mendekomposisi Citra Awal

Gambar 3.12 Menunjukkan citra yang di dekomposisi menjadi empat bagian frekuensi.

2. Lakukan kembali DWT sehingga mendapatkan 8 frekuensi HL_2, LH_2, HL_2, LH_2



Gambar 3.13 Mendekomposisi Citra Awal Pada Frekuensi LL_1

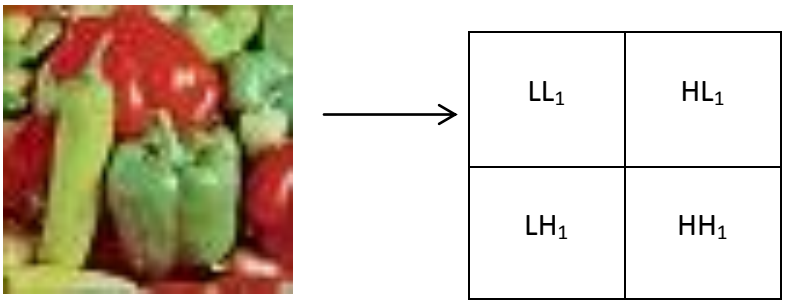
Gambar 3.13 menunjukkan citra yang didekomposisi menjadi 8 bagian frekuensi.

3. Optimasi menggunakan genetika untuk menyisipkan *watermark*

4. Lakukan DCT pada LL_1

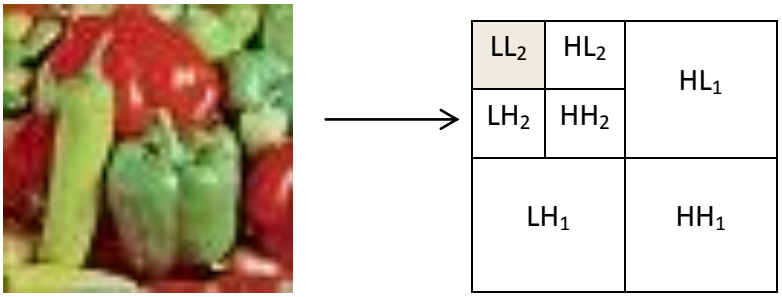
5. Lakukan SVD pada LL_1
   $W= U\ S\ V^T$

6. Lakukan DWT pada watermark



Gambar 3.14 Mendekomposisi Citra Watermark

Gambar 3.14 Menunjukkan Citra *Watermark* Yang Didekomposisi Menjadi Empat Bagian Frekuensi.



Gambar 3.15 Mendekomposisi Citra *Watermark* Pada Frekuensi LL_1

Gambar 3.15 menunjukka citra *watermark* yang didekomposisi menjadi delapan frekuensi.

7. Lakukan DCT pada bagian frekuensi LH_2 dari *watermark*

8. Lakukan SVD pada bagian frekuensi LH_2 dari *watermark*

9. Lakukan invers DCT

10. Lakukan invers DWT

3.1.6 Analisis Metode DWT

Proses *watermarking* menggunakan DWT yaitu citra digital sebelumnya didekomposisi menggunakan DWT untuk dapat menyisipkan *watermark*, kemudian lakukan IDWT untuk mendapatkan citra ber-*watermark*. Penyisipan watermark dilakukan dengan cara memodifikasi koefisien pada rentang frekuensi LL, LH, HL atau HH yang merupakan rentang frekuensi hasil dekomposisi citra. Algoritma umum penyisipan watermark pada koefisien rentang frekuensi adalah

$$f' = f + \alpha.w(k) + K(k), k = 1$$

Dimana α merupakan kekuatan penyisipan yang mengontrol tingkat kekuatan penyisipan *watermark* dan f ' adalah koefisien sinyal asal yang telah dimodifikasi. Penyisipan *watermark* pada citra digital menggunakan DWT ini dilakukan pada koefisien rentang frekuensi (koefisien DWT) sebelum direkonstruksi menggunakan IDWT untuk menjadi citra ber*watermark*. Penyisipan citra watermark diawali dengan mengubah susunan citra watermark ke dalam rangkaian matriks. Selanjutnya penyisipan *watermark* dilakukan dengan langkah-langkah:

1. Setelah citra host didekomposisi dalam l tingkatan DWT, watermark disisipkan kedalam rentang frekuensi LHl atau HLl , (l∈{2,3,4}).

2. Mencari koefisien terbesar fLH dari dari rentang frekuensi LH atau koefisien terbesar fHL dari rentang frekuensi HL.

3. Menyisipkan w ke dalam rentang frekuensi LH atau HL dengan persamaan: f' LH(m,n)=f LH(m,n)+α.w(m,n)+K(m,n);m,n=1,...,L (9) f'HL(m,n)=fHL(m,n)+α.w(m,n)+K(m,n);m,n=1,...,L (10) Dimana LH f (m,n) merupakan koefisien terbesar yang dipilih dan LH f ′ (m,n)

merupakan koefisien yang dimodifikasi pada posisi (m,n) untuk rentang frekuensi LH. HL f (m,n) merupakan koefisien terbesar yang dipilih dan HL f HL f ′ (m,n) merupakan koefisien yang dimodifikasi pada posisi (m,n) untuk rentang frekuensi HL. α, seperti yang telah dijelaskan di dalam bab Dasar Teori, merupakan kekuatan penyisipan *watermark* atau dapat dikatakan sebagai faktor skala persentase dari citra host dan citra watermark pada citra ber-watermark yang dibentuk. Bentuk kunci K ini adalah sebuah array yang memiliki panjang tertentu dengan panjang maksimal adalah sepanjang *watermark*. Kunci K ditambahkan dalam proses penyisipan *watermark* mealalui proses penjumlahan terhadap nilai watermark pada indeks yang bersesuaian.

4. Menerapkan *Inverse Discrete Wavelet Transform* (IDWT) untuk membentuk citra ber-*watermark.*

   Perhitungan menggunakan teknik DWT citra yang digunakan adalah bunga.jpg. Matriks yang digunakan berukuran 4 x 4 nilai yang diambil dari matriks lenna.

$$A = \begin{bmatrix} 5 & 11 & 8 & 10 \\ 9 & 8 & 4 & 12 \\ 1 & 10 & 11 & 4 \\ 19 & 6 & 15 & 7 \end{bmatrix}$$

   Dilakukan DWT 2level maka matriks A dikalikan 2

$$L = 2.\begin{bmatrix} 5 & 11 & 8 & 10 \\ 9 & 8 & 4 & 12 \\ 1 & 10 & 11 & 4 \\ 19 & 6 & 15 & 7 \end{bmatrix}$$

$$L = \begin{bmatrix} 10 & 22 & 16 & 20 \\ 18 & 16 & 4 & 24 \\ 2 & 20 & 22 & 8 \\ 36 & 12 & 30 & 14 \end{bmatrix}$$

$$LL = \begin{bmatrix} 10 & 22 & 16 & 20 \\ 18 & 16 & 4 & 24 \\ 2 & 20 & 22 & 8 \\ 36 & 12 & 30 & 14 \end{bmatrix} * \begin{bmatrix} 10 & 22 & 16 & 20 \\ 18 & 16 & 4 & 24 \\ 2 & 20 & 22 & 8 \\ 36 & 12 & 30 & 14 \end{bmatrix}$$

LL00 = (10 * 10) + (22 * 18) + (16*2) + (20 * 12)

= 100 + 396 + 32 + 720 = 1284

LL01 = (10 * 22) + (22 * 16) + (16*20) + (20 * 12)

= 220 + 352 + 320 + 720 = 1132

LL02 = (10 * 16) + (22 * 8) + (16*22) + (20 * 30)

= 160 + 396 + 32 + 720 = 1288

LL03 = (10 * 20) + (22 * 24) + (16*8) + (20 * 14)

= 200 + 528 + 128 + 280 = 1136

LL10 = (18 * 10) + (16 * 18) + (8*2) + (24 * 36)

= 180 + 288 + 16 + 864 = 1348

LL11 = (18 * 22) + (16 * 16) + (8*20) + (24 * 12)

= 396 + 256 + 160 + 288 = 1100

LL12 = (18 * 16) + (16 * 18 + (8*22) + (24 * 30)

= 288 + 128 +176 + 720 = 1312

LL13 = (18 * 20) + (16 * 24) + (8*8) + (24 * 14)

= 360 + 384 + 64 + 336 = 1144

LL20 = (2 * 10) + (20 * 18) + (22*2) + (8 * 36)

= 20 + 360 + 44 + 288 = 712

LL21 = (2 * 22) + (20 * 16) + (22*20) + (8 * 12)

= 44 + 320 + 440 + 96 = 900

LL22 = (2 * 16) + (20 * 8) + (22*22) + (8 * 30)

= 32 + 160 + 484 + 240 = 916

LL23 = (2 * 20) + (20 * 24) + (22*8) + (8 * 14)

= 40 + 480 + 176 + 112 = 808

LL30 = (36 * 10) + (12 * 18) + (30*2) + (14 * 36)

= 360 + 216 + 60 + 504 = 1140

LL31 = (36 * 22) + (12 * 16) + (30*20) + (14 * 12)

= 792 + 192 + 600 + 168 = 1752

LL32 = (36 * 16) + (12 * 8) + (30*22) + (14 * 30)

=576 + 96 + 660 + 420 = 1752

LL33 = (36 * 20) + (12 * 24) + (30*8) + (14 * 14)

= 720 + 288 + 240 + 196 = 1444

Maka didapatkan

$$LL = \begin{bmatrix} 1284 & 1132 & 1288 & 1136 \\ 1348 & 1100 & 1312 & 1144 \\ 712 & 900 & 916 & 808 \\ 1140 & 1752 & 1752 & 1444 \end{bmatrix}$$

3.1.7 Analisis Metode DWT dengan pendekatan algoritma genetika

1. Proses inisialisasi

   Pada proses inisialisasi menggunakan 4 buah populasi pada generasi pertama

   Kromosom[1] = LL(m,n) = 5

   Kromosom[2] = HL(m,n) = 11

   Kromosom[3] = LH(m,n) = 9

   Kromosom[4] = HH(m,n) = 8

2. Evaluasi kromosom

   Kromosom[1] : fitness[1] = fLL(m,n) + α.w(m,n) + k(m,n)

   Kromosom[2] : fitness[2] = fHL(m,n) + α.w(m,n) + k(m,n)

   Kromosom[3] : fitness[3] = fLH(m,n) + α.w(m,n) + k(m,n)

Kromosom[4] : fitness[4] = fHH(m,n) + α.w(m,n) + k(m,n)

3. Seleksi kromosom

   Pada tahapan seleksi kromosom yang dicari adalah nilai fitness yang terkecil dengan menggukan rumus Q = 1/fitness. Setelah mendapatkan nilai fitness terkecil maka pada frekuensi itulah akan disisipi *watermark.*

Dalam *watermarking* citra digital, populasi diinisialkan dengan memilih posisi acak dalam citra awal dan menyisipkan citra watermark pada posisi yang yang dipilih. Untuk meningkatkan ketahanan supaya tahan dari serangan maka nilai α harus dioptimalkan.

Fungsi fitness = PSNR + 100 * ρ

Fungsi fitness dievaluasi untuk semua individu dalam populasi dan paling cocok individu bersama dengan sesuai nilai fitness diperoleh. Operator genetik seperti crossover dan mutasi yang dilakukan pada orang tua yang dipilih untuk menghasilkan keturunan baru yang termasuk dalam populasi membentuk generasi berikutnya. Seluruh proses diulang untuk beberapa generasi sampai solusi terbaik diperoleh. Itu faktor korelasi ρ mengukur kesamaan antara asli *watermark* dan *watermark* yang diekstraksi dari menyerang *image watermark* (ketahanan).

Hasil generasi yang dibangkitkan dapat dilihat pada tabel 3.1.

Tabel 3.1 Gen Yang Dibangkitkan

Citra	Gen	MSE	PSNR	NC	Fitness
koala	10	5.12322	41.03422	0.988	**53.5273**
	20	5.1156	41.03233	0.9812	52.0364
	30	5.0122	41.13028	0.9823	52.9736
	40	5.2366	40.76901	0.9848	52.8790
beer	10	5.9415	40.39112	0.9965	**52.3499**
	20	6.1276	40.2622	0.9982	52.2403
	30	6.6651	39.9823	0.9952	51.7612
	40	6.9429	39.9823	0.9971	51.3392
cup	10	4.2378	41.2832	0.9638	53.5409
	20	4.4839	42.0812	0.9217	**53.6582**
	30	4.1293	41.6725	0.9926	52.9837
	40	4.8342	41.0931	0.9632	53.2871

Hasil populasi yang didapat setelah generasi dibangkitkan dapat dilihat pada tabel 3.2.

Table 3.2 Hasil Populasi Yang Didapat Setelah Generasi Dibangkitkan

Citra	Populasi	MSE	PSNR	NC	Fitness
koala	10	5.12322	41.03422	0.988	**53.5273**
	20	5.1156	41.03233	0.9812	52.0364
	30	5.0122	41.13028	0.9823	52.9736
	40	5.2366	40.76901	0.9848	52.8790
beer	10	5.9415	40.39112	0.9965	**52.3499**
	20	6.1276	40.2622	0.9982	52.2403
	30	6.6651	39.9823	0.9952	51.7612
	40	6.9429	39.9823	0.9971	51.3392
cup	10	4.2378	41.2832	0.9638	53.5409
	20	4.4839	42.0812	0.9217	**53.6582**
	30	4.1293	41.6725	0.9926	52.9837
	40	4.8342	41.0931	0.9632	53.2871

Hasil cromosom yang dihasilkan dapat dilihat pada tabel 3.3.

Tabel 3.3 Hasil Cromosom Yang Didapat

Citra	cro	MSE	PSNR	NC	Fitness
koala	10	5.12322	41.03422	0.988	**53.5273**
	20	5.1156	41.03233	0.9812	52.0364
	30	5.0122	41.13028	0.9823	52.9736
	40	5.2366	40.76901	0.9848	52.8790
beer	10	5.9415	40.39112	0.9965	**52.3499**
	20	6.1276	40.2622	0.9982	52.2403
	30	6.6651	39.9823	0.9952	51.7612
	40	6.9429	39.9823	0.9971	51.3392
cup	10	4.2378	41.2832	0.9638	53.5409
	20	4.4839	42.0812	0.9217	**53.6582**
	30	4.1293	41.6725	0.9926	52.9837
	40	4.8342	41.0931	0.9632	53.2871

Hasil mutasi yang dihasilkan dapat dilihat pada tabel 3.4.

Tabel 3.4 Hasil Mutasi Yang Didapatkan

Citra	Mutasi	MSE	PSNR	NC	Fitness
koala	10	5.12322	41.03422	0.988	**53.5273**
	20	5.1156	41.03233	0.9812	52.0364
	30	5.0122	41.13028	0.9823	52.9736
	40	5.2366	40.76901	0.9848	52.8790
beer	10	5.9415	40.39112	0.9965	**52.3499**
	20	6.1276	40.2622	0.9982	52.2403
	30	6.6651	39.9823	0.9952	51.7612
	40	6.9429	39.9823	0.9971	51.3392
cup	10	4.2378	41.2832	0.9638	53.5409
	20	4.4839	42.0812	0.9217	53.6582
	30	4.1293	41.6725	0.9926	**52.9837**
	40	4.8342	41.0931	0.9632	53.2871

3.1.8 Analisis Metode DCT

Misalkan ada sebuah citra X yang akan di *watermark* yang memiliki format .jpg. Karena komputer hanya dapat mengolah data yang bersifat numerik, supaya citra tersebut dapat diproses oleh komputer maka harus direpresentasikan secara numerik yang merupakan angka angka yang menunjukkan jumlah intensitas warna pada citra tersebut. Proses pengubahan citra menjadi matriks yaitu citra disimpan oleh komputer dengan titik titik berupa koordinat yang disebut dengan pixel. Kemudian citra akan dikomposisi menggunakan DCT misalkan citra asli dengan matriks X dan *watermark* dengan matriks W berukuran 2 x 2 pixel, maka penyisipan *watermark* W kedalam citra X dilakukan dengan mendekomposisi citra X

menjadi matriks. Tahap selanjutnya adalah proses *watermarking*, menggunakan tiga elemen matriks hasil representasi citra sampel, yaitu matriks red (R), matriks green (G) dan matriks blue (B). Kemudian untuk membuat citra ber-watermark X_w, mendekomposisikan citra menjadi matriks U, S, dan V untuk mendapatkan nilai singular dari citra. Matriks yang digunakan berukuran 8 x 8 yang didapatkan dari matriks paprika.

$$A = \begin{bmatrix} 135 & 135 & 132 & 134 & 136 & 131 & 132 & 129 \\ 137 & 135 & 132 & 134 & 135 & 132 & 132 & 129 \\ 131 & 132 & 132 & 132 & 134 & 130 & 128 & 128 \\ 127 & 131 & 132 & 128 & 132 & 131 & 125 & 122 \\ 128 & 130 & 128 & 127 & 129 & 128 & 129 & 126 \\ 128 & 131 & 127 & 120 & 130 & 128 & 127 & 129 \\ 128 & 127 & 128 & 128 & 129 & 128 & 126 & 128 \\ 129 & 129 & 128 & 128 & 129 & 128 & 127 & 126 \end{bmatrix}$$

DCT mengerjakan nilai pixel antara -128 ke 127 maka matrik akan dikurangi 128 dari setiap masukannya sehingga menjadi

$$= \begin{bmatrix} 7 & 7 & 4 & 6 & 8 & 3 & 4 & 1 \\ 7 & 7 & 4 & 6 & 7 & 4 & 4 & 1 \\ 9 & 4 & 4 & 4 & 6 & 2 & 0 & 0 \\ 3 & 3 & 4 & 0 & 4 & 3 & -3 & -6 \\ -1 & 2 & 0 & -1 & 1 & 0 & -1 & 1 \\ 0 & 3 & -1 & -8 & 2 & 0 & -1 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 & -2 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & -1 & -2 \end{bmatrix}$$

Selanjutnya penyusunan kembali citra

$$hasil = \begin{bmatrix} 134 & 134 & 133 & 132 & 132 & 131 & 130 & 130 \\ 134 & 133 & 133 & 132 & 131 & 131 & 130 & 130 \\ 133 & 133 & 132 & 132 & 131 & 131 & 130 & 129 \\ 132 & 132 & 131 & 131 & 139 & 139 & 129 & 129 \\ 121 & 131 & 131 & 130 & 129 & 129 & 128 & 128 \\ 131 & 130 & 130 & 129 & 128 & 128 & 127 & 127 \\ 130 & 130 & 129 & 129 & 128 & 127 & 127 & 127 \\ 130 & 130 & 129 & 128 & 128 & 127 & 126 & 126 \end{bmatrix}$$

3.1.9 Analisis Metode SVD

Singular Value Decomposition (SVD) merupakan teknik analisis numerik dalam mendiagonalkan suatu matriks. Watermarking dengan menggunakan SVD terdapat tiga langkah yaitu

1. Melakukan SVD pada citra asli A dengan $A = U\sum V^T$
2. Menambahkan citra watermark W ke $\sum$ dengan $\sum_n = \sum + \alpha W$
3. Mendapatkan citra yang ber-watermark A_w dengan $A_w = U\sum_n V^T$

Citra lena yang sudah dalam bentuk sebuah matrik A dengan ukuran 4 x 2 nilai yang didapatkan dari matriks lenna $A = \begin{bmatrix} 76 & 94 \\ 86 & 91 \\ 61 & 79 \\ 47 & 61 \end{bmatrix}$

Maka

$$A.A^T = \begin{bmatrix} 76 & 94 \\ 86 & 91 \\ 61 & 79 \\ 47 & 61 \end{bmatrix} \begin{bmatrix} 76 & 86 & 61 & 47 \\ 94 & 91 & 79 & 61 \end{bmatrix}$$

$$= \begin{bmatrix} 14612 & 15090 & 12062 & 9306 \\ 15090 & 15677 & 12435 & 9593 \\ 12062 & 12435 & 9962 & 7686 \\ 9306 & 9593 & 7686 & 5930 \end{bmatrix}$$

Setelah itu cari eigenvaluenya

$$\begin{bmatrix} 14612-\lambda & 15090 & 12062 & 9306 \\ 15090 & 15677-\lambda & 12435 & 9593 \\ 12062 & 12435 & 9962-\lambda & 7686 \\ 9306 & 9593 & 7686 & 5930-\lambda \end{bmatrix} x = (W - \lambda I)x = 0$$

kolom pertama matriks U

$$U = \begin{bmatrix} 0.82 & -0.58 & 0 & 0 \\ 0.58 & 0.82 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A^T.A akan mengisi kolom dari matriks V maka dilakukan analisis yang sama

$$A^T.A = \begin{bmatrix} 76 & 86 & 61 & 47 \\ 94 & 91 & 79 & 61 \end{bmatrix} \begin{bmatrix} 76 & 94 \\ 86 & 91 \\ 61 & 79 \\ 47 & 61 \end{bmatrix}$$

Sehingga didapatkan

$$V = \begin{bmatrix} 40 & 91 \\ 91 & 40 \end{bmatrix}$$

S merupakan akar kuadrat dari A^T.A dan didapatkan

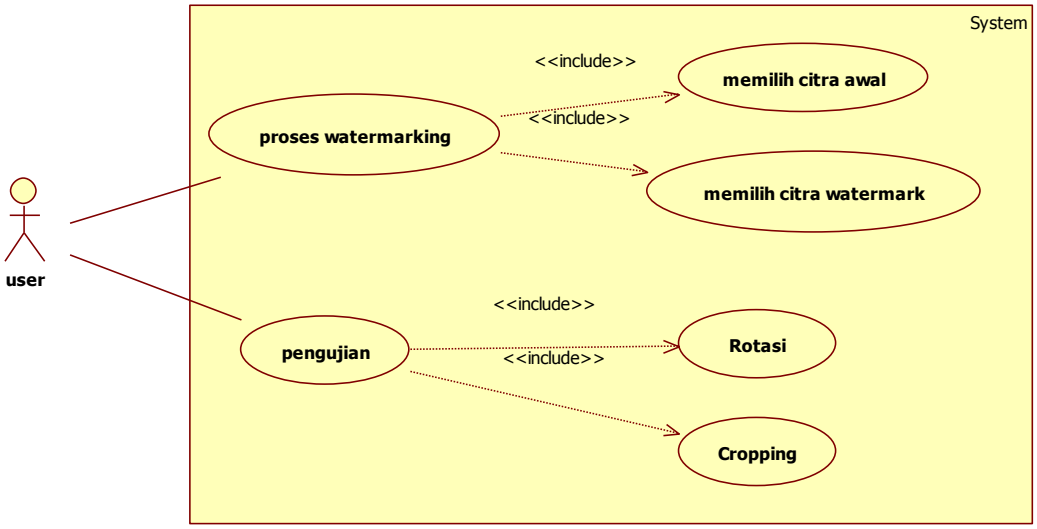
$$S = \begin{bmatrix} 65 & 0 \\ 0 & 51 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

3.2 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan jenis kebutuhan proses-proses yang akan dilakukan oleh suatu sistem dan memberikan informasi tentang apa saja yang dihasilkan dari sistem tersebut. *Tools* yang digunakan adalah *use case diagram, activity diagram, class diagram* dan *sequence diagram*.

3.2.1 Use Case Diagram

Use case diagram merupakan model fungsionalitas dari sebuah sistem. Menggambarkan suatu interaksi antara aktor dengan sistem. *Use case* diagram *watermarking* dapat dilihat pada gambar 3.15.



Gambar 3.16 *Use Case* Diagram *Watermarking*

3.2.2 Skenario *Use Case*

Skenario *use case* menunjukkan proses apa saja yang terjadi pada setiap bagiannya, yang merupakan penjelasan yang lebih rinci dari setiap fungsionalitasnya. Sistem akan merespon apa yang diperintahkan oleh user.

1. Skenario Proses *Watermarking*

   Skenario *use case* proses *watermarking* dapat dilihat pada tabel 3.5

Tabel 3.5 Skenario *Use Case* Proses *Watermarking*

Usecase name	**Proses *Watermarking***	
Related Requirement	SKPL-F-1	
Goal In Context	aplikasi melakukan proses *watermarking*	
Preconditions	user harus memilih citra awal dan citra *watermark*	
Succesfull End Condition	aplikasi berhasil melakukan proses *watermarking*	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan menu *watermarking*	
Main Flow	***Step***	***Action***
	1	*User* membuka aplikasi
	2	*User* memilih citra awal
	3	*User* memilih citra *watermark*
	4	*User* memilih proses *watermarking*
	5	Melakukan proses DWT
	6	Melakukan proses DCT
	7	Melakukan proses SVD

2. Skenario Memilih Citra Awal

   Skenario *use case* memilih citra awal dapat dilihat pada tabel 3.6.

Tabel 3.6 Skenario *Usecase* Memilih Citra Awal

Usecase name	**Memilih Citra Awal**	
Related Requirement	SKPL-F-2	
Goal In Context	aplikasi memilih citra awal	
Preconditions	user harus memilih citra awal	
Succesfull End Condition	aplikasi berhasil memilih citra awal	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan hasil pilihan citra awal	
Main Flow	***Step***	***Action***
	1	*User* memilih citra awal

3. Skenario Memilih Citra Watermark

   Skenario memilih citra watermark dapat dilihat pada tabel 3.7.

Tabel 3.7 Skenario *Use Case* Memilih Citra *Watermark*

Usecase name	**Memilih Citra Awal**	
Related Requirement	SKPL-F-3	
Goal In Context	aplikasi memilih citra *watermark*	
Preconditions	user harus memilih citra *watermark*	
Succesfull End Condition	aplikasi berhasil memilih citra *watermark*	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan hasil pilihan citra watermark	
Main Flow	***Step***	***Action***
	1	User memilih citra *watermark*

4. Skenario Pengujian

   Skenario use case pengujian dapat dilihat pada tabel 3.8.

Tabel 3.8 Skenario *Use Case* Pengujian

Usecase name	**Pengujian**	
Related Requirement	SKPL-F-4	
Goal In Context	aplikasi melakukan pengujian	
Preconditions	user harus memilih pengujian rotasi dan *cropping*	
Succesfull End Condition	aplikasi berhasil melakukan pengujian	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan hasil pengujian	
Main Flow	***Step***	***Action***
	1	*User* memilih pengujian rotasi
	2	*User* memilih pengujian *cropping*

5. Skenario Rotasi

   Skenario use case rotasi dapat dilihat pada tabel 3.9.

Tabel 3.9 Skenario *Use Case* Rotasi

Usecase name	**Rotasi**	
Related Requirement	SKPL-F-5	
Goal In Context	aplikasi memilih rotasi	
Preconditions	user harus memilih rotasi	
Succesfull End Condition	aplikasi berhasil melakukan serangan dengan rotasi	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan hasil rotasi	
Main Flow	***Step***	***Action***
	1	User memilih rotasi

6. Skenario *cropping*

   Skenario *use case cropping* dapat dilihat pada tabel 3.10.

Tabel 3.10 Skenario Use Case Cropping

Usecase name	***Cropping***	
Related Requirement	SKPL-F-6	
Goal In Context	aplikasi memilih *cropping*	
Preconditions	user harus memilih *cropping*	
Succesfull End Condition	aplikasi berhasil melakukan serangan berupa *cropping*	
Failed End Condition	*None*	
Primary Actory	*user*	
Secondary Actory	*None*	
Trigger	Aplikasi menampilkan hasil cropping	
Main Flow	***Step***	***Action***
	1	*User* memilih *cropping*

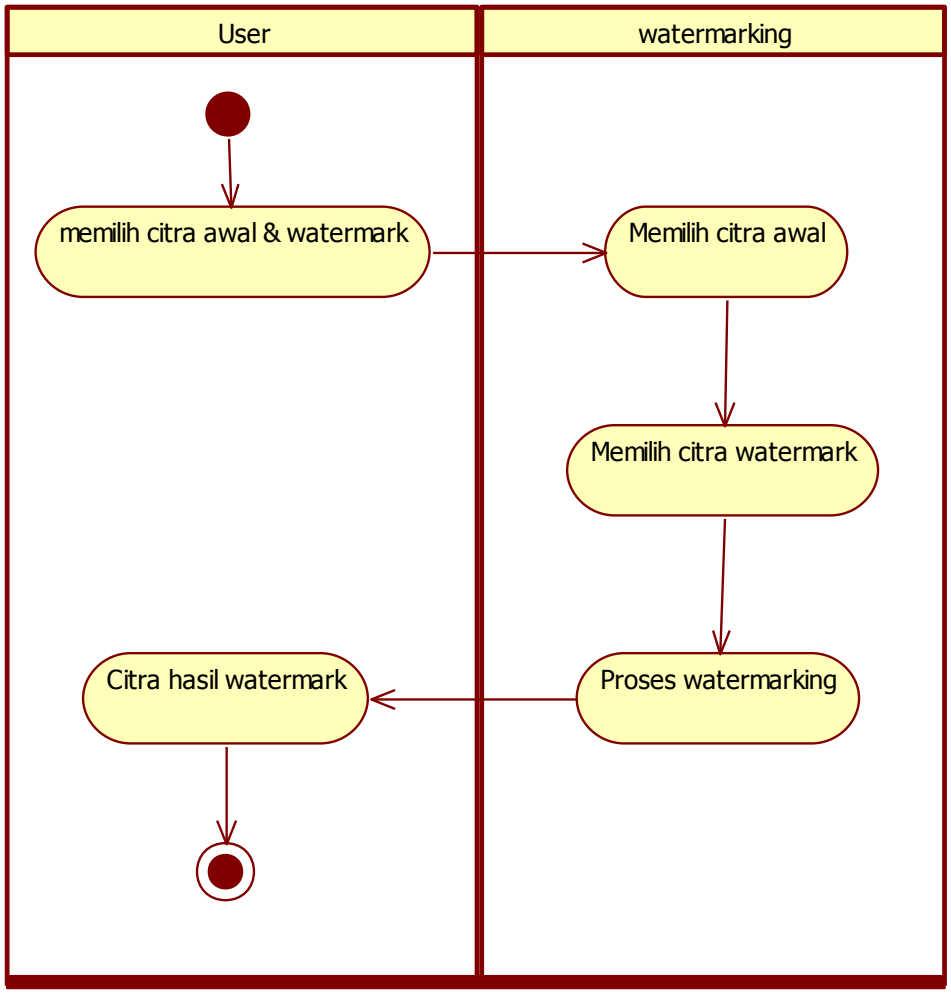
SKPL-ID dapat dilihat pada tabel 3.11.

Tabel 3.11 SKPL-ID

SKPL-ID	**Keterangan**
SKPL-F-1	Melakukan proses *watermarking*
SKPL-F-2	Memilih citra awal
SKPL-F-3	Memilih citra *watermark*
SKPL-F-4	Melakukan pengujian
SKPL-F-5	Memilih rotasi
SKPL-F-6	Memilih *cropping*

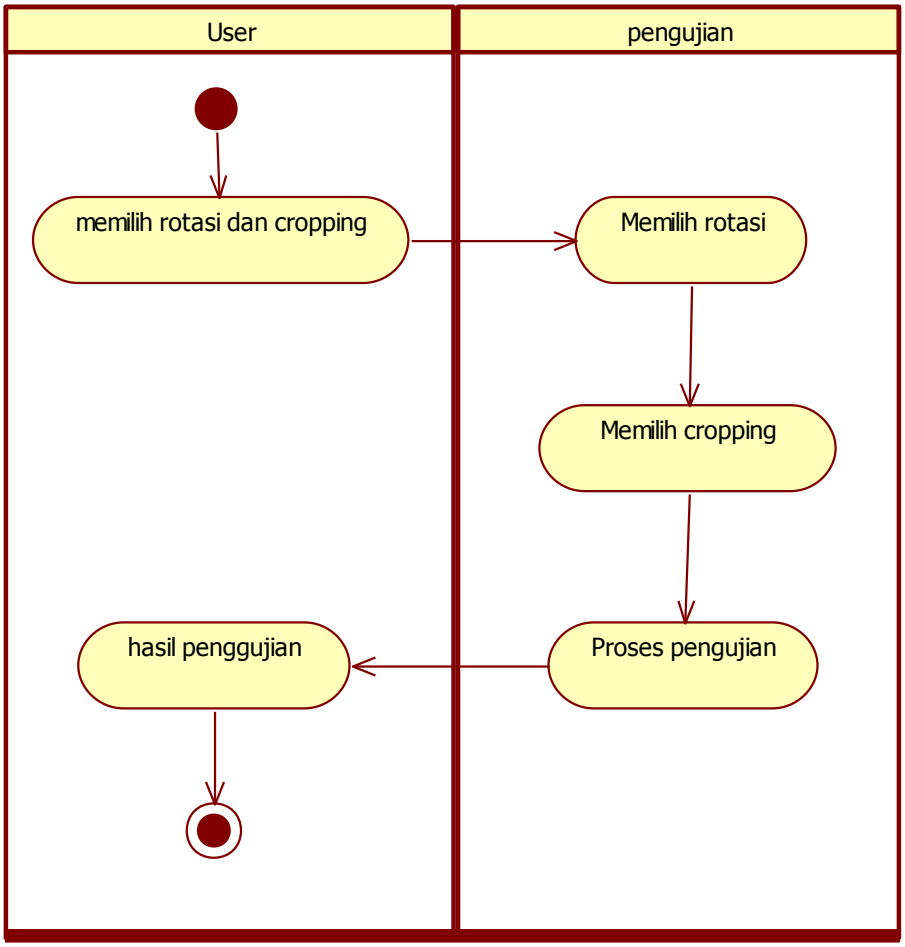
3.2.3 Activity Diagram

Activity diagram menggambarkan bagaimana urutan urutan aktivitas dalam suatu proses. *Activity diagram watermarking* ditunjukkan pada gambar 3.12 dan activity diagram pengujian ditunjukkan pada gambar 3.13.



Gambar 3.17 *Activity Diagram* watermarking

Activity diagram watermarking menunjukkan bagaimana urutan urutan dalam proses watermarking. Seperti yang terlihat pada gambar 3.17 user harus memilih citra awal kemudian memilih citra *watermark* setelah itu melakukan proses *watermarking*.

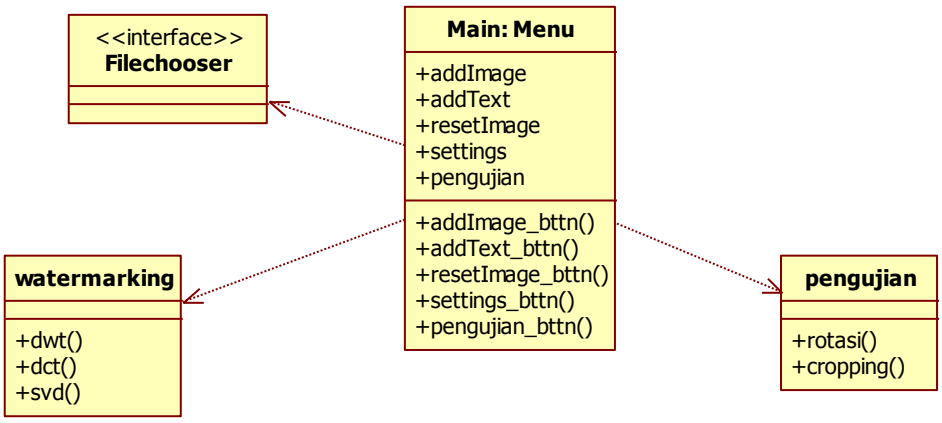


Gambar 3.18 *Activity diagram* pengujian

Activity diagram pengujian menunjukkan urutan urutan yang ada dalam proses pengujian. Seperti pada gambar 3.18 user memilih rotasi dan *cropping* kemudian melakukan proses pengujian.

3.2.4 Class Diagram

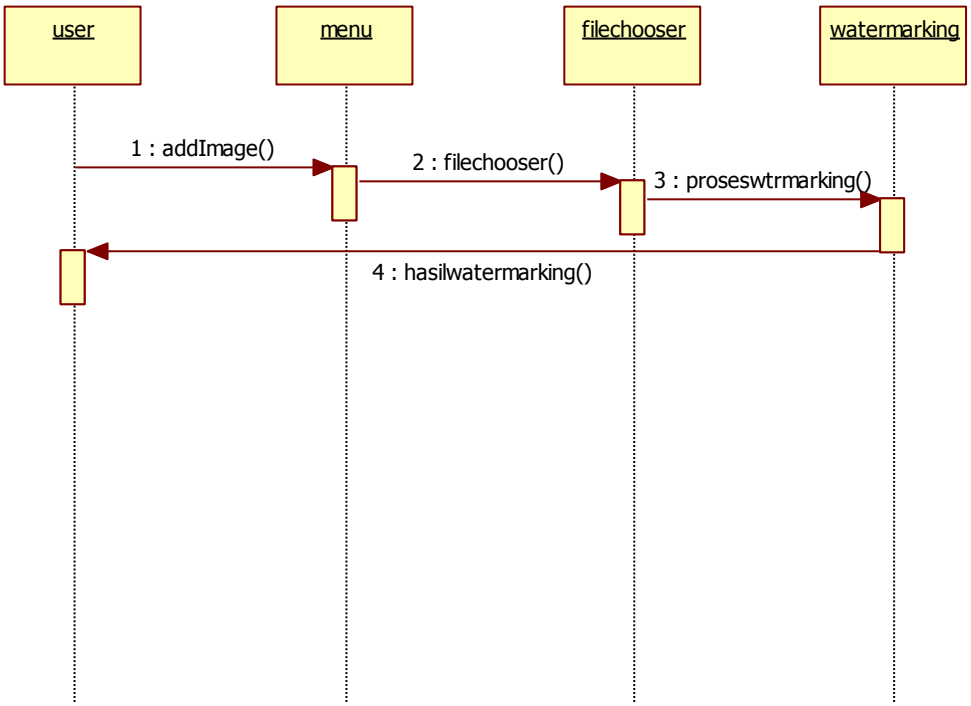
Class diagram digunakan untuk memberi gambaran suatu sistem. Dapat dilihat dari class class yang ada dan bagaimana relasinya. Class diagram pada apliakasi *watermarking* dapat dilihat pada gambar 3.19.



Gambar 3.19 *Class Diagram*

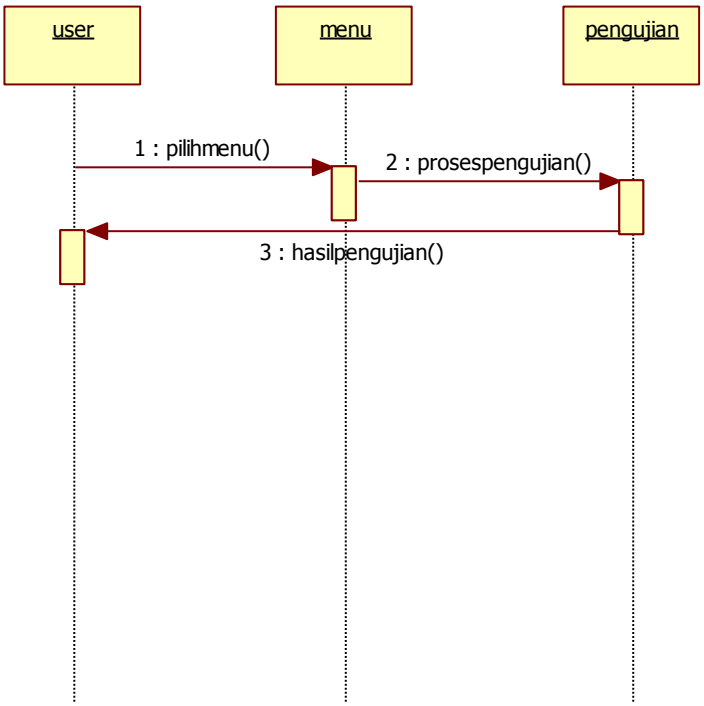
3.2.5 Sequence Diagram

Sequence diagram merupakan gambaran antara interaksi antar objek objeknya. Merupakan langkah langkah dari beberarapa event untuk mendapatkan suatu output tertentu. **Sequence diagram** *watermarking* dapat dilihat pada gambar 3.20 dan sequence diagram pengujian dapat dilihat pada gambar 3.21.



Gambar 3.20 Sequence Diagram *Watermarking*

Sequence diagram *watermarking* menjelaskan interaksi antara user dengan sistem dalam proses *watermarking* seperti pada gambar 3.20.

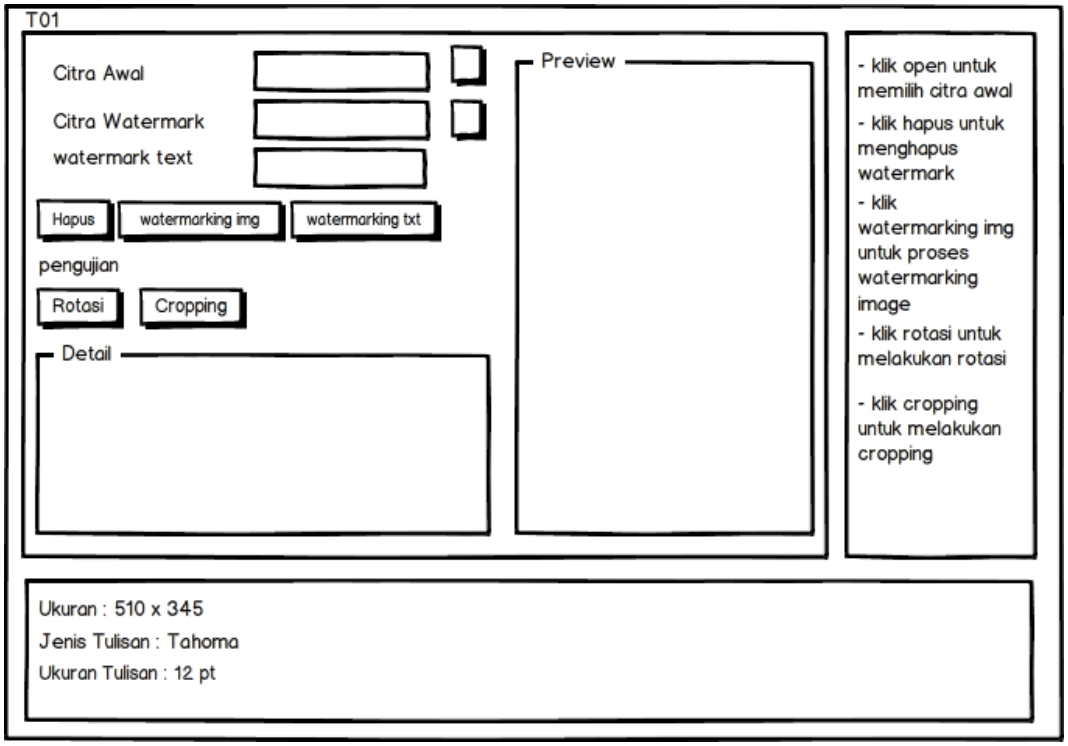


Gambar 3.21 *Sequence Diagram* Pengujian

Sequence diagram pengujian menjelaskan bagaimana interaksi antara user dan sistem ketika proses pengujian seperti pada gambar 3.21.

3.2.6 Perancangan Antarmuka

Perancangan antarmuka merupakan tahapan pembuatan rancangan pada aplikasi *watermarking* yang akan dibuat. Untuk perancangan antarmuka *watermarking* dapat dilihat pada gambar 3.17



Gambar 3.22 Perancangan Antarmuka

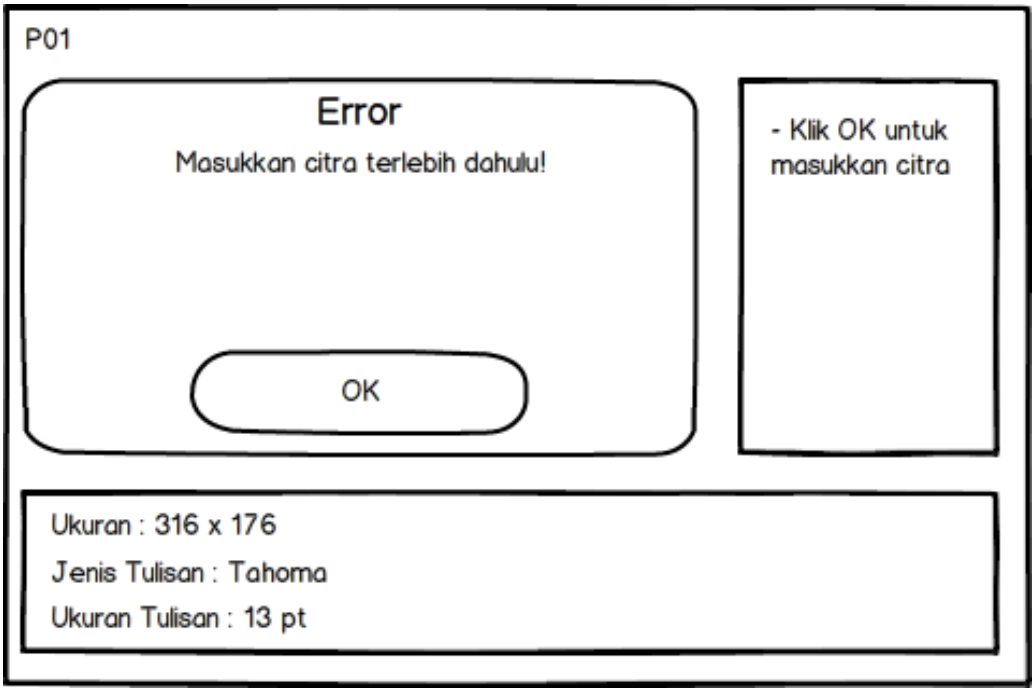
Gambar 3.22 menunjukkan perancangan antarmuka yang akan dibangun. User akan memilih button open untuk memilih citra awal dan memilih citra *watermark* kemudian user memilih tombol *watermarking image* untuk proses *watermarking*. User juga dapat menyisipkan text dengan menuliskan teksnya pada textbox dan menekan tombol add text untuk menyisipkannya.

3.2.7 Perancangan Pesan

Perancangan pesan adalah rancangan pesan yang akan dibuat pada aplikasi. Pesan akan memberikan suatu informasi kepada user. Perancangan pesan yang ada pada aplikasi ini adalah:

1. Perancangan pesan jika belum memilih citra

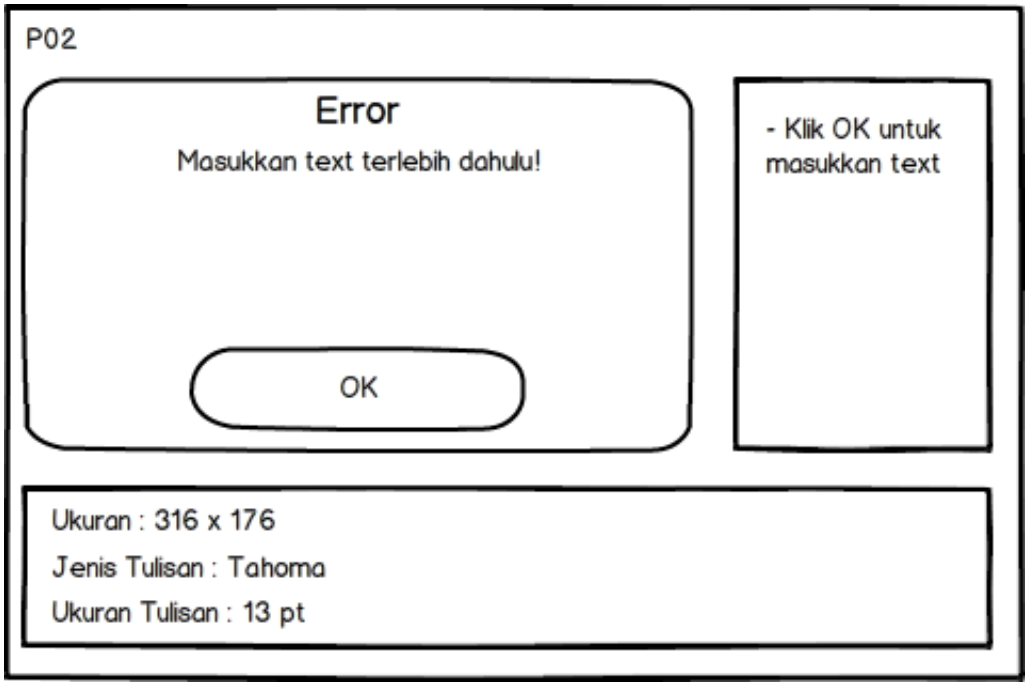
   Pesan ini akan muncul jika *user* menekan tombol *add image* namun belum memilih citra awal dan citra *watermark*.



Gambar 3.23 Perancangan Pesan belum memilih citra

2. Perancangan pesan jika belum memilih teks

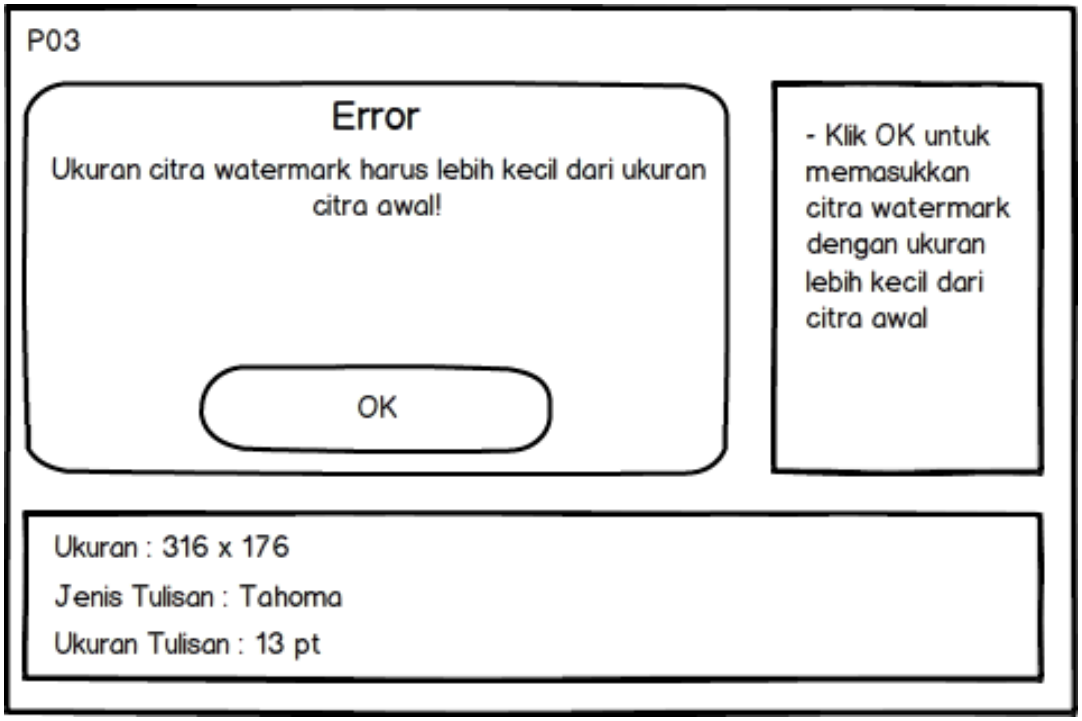
   Pesan ini akan muncul jika user menekan tombol add text namun belum mengetikkan text yang akan disisipi.



Gambar 3.24 Perancangan Pesan Belum Memasukkan Teks

3. Perancangan pesan jika ukuran citra watermark lebih besar daripada citra awal

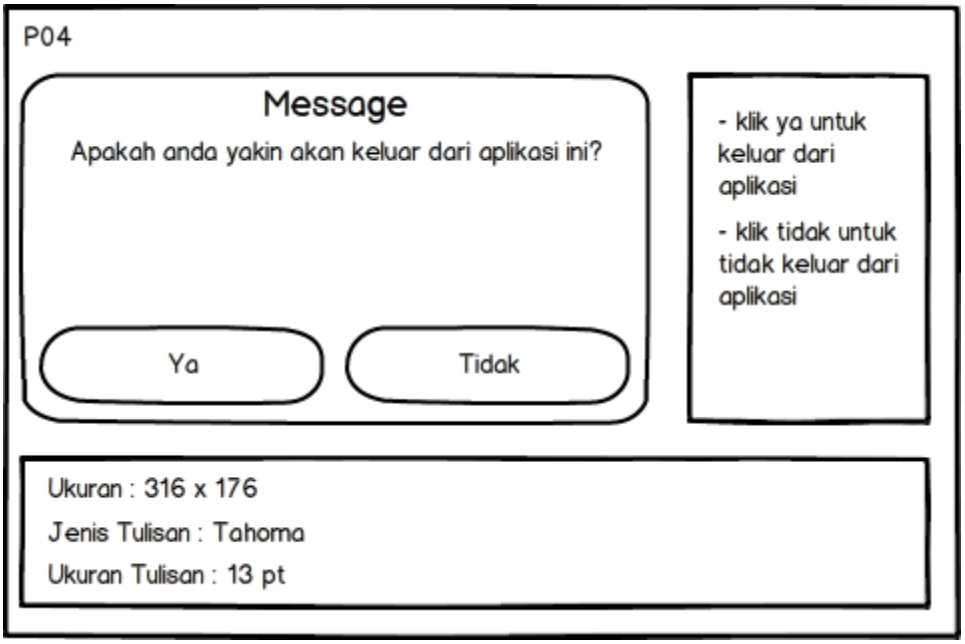
   Pesan ini akan muncul ketika user memilih citra watermark yang ukurannya lebih besar daaripada citra awal.



Gambar 3.25 Perancangan pesan jika ukuran citra watermark lebih besar dari citra awal

4. Perancangan pesan jika akan keluar dari aplikasi

   Pesan ini akan muncul ketika user menekan tombol close.



Gambar 3.26 Perancangan pesan jika akan keluar dari aplikasi