Robust drift-free open-loop H.264 watermarking

Wei Chen Zafar Shahid Thomas Stütz
Florent Autrusseau Patrick Le Callet

This document gives the "*Mathematica*" functions that are used to determine the optimal solutions of the system of linear equations to be used in paper "Robust drift-free open-loop H.264 watermarking".

All the "*Mathematica*" functions are given in the code inset below, and next, we describe each function.

Note: *"Mathematica"* can be found here: http://www.wolfram.com/mathematica/ or, as an alternative, *"mathics"* could be used as well: http://www.mathics.net.

```
m=4;
mat2vec[m_]:=Flatten[m];
vec2m[v_]:={v[[1;;4]],v[[5;;8]],v[[9;;12]],v[[13;;16]]};
MatrixForm[cM = {{1,1,1,1},{1,1/2, -1/2,
    -1},{1,-1,-1,1},{1/2,-1,1,-1/2}}];
MatrixForm[cT = Transpose[cM]];
MatrixForm[eI={{a^2, a*b, a^2, a*b}, {a*b, b^2, a*b, b^2}, {a^2, a*
    b, a^2, a*b}, {a*b, b^2,a*b, b^2}}];
a=1/2;  b=Sqrt[2/5];
MatrixForm[eI];
MatrixForm[Y = {{y00, y01, y02, y03}, {y10, y11, y12, y13}, {y20,
    y21, y22, y23}, {y30, y31, y32, y33}}];
pM = cT.(Y).cM;
r1 = Expand[pM[[1,4]]];
e1={1,-1,1,-1/2,1,-1,1,-1/2,1,-1,1,-1/2,1/2,-1/2,1/2,-1/4};
r2=Expand[pM[[2,4]]]
e2={1,-1,1,-1/2,1/2,-1/2,1/2,-1/4,-1,1,-1,1/2,-1,1,-1,1/2};
r3=Expand[pM[[3,4]]]
e3={1,-1,1,-1/2,-1/2,1/2,-1/2,1/4,-1,1,-1,1/2,1,-1,1,-1/2};
r4=Expand[pM[[4,4]]];
e4={1,-1,1,-1/2,-1,1,-1,1/2,1,-1,1,-1/2,-1/2,1/2,-1/2,1/4};
r5=Expand[pM[[4,1]]];
e5={1,1,1,1/2,-1,-1,-1,-1/2,1,1,1,1/2,-1/2,-1/2,-1/2,-1/4};
r6=Expand[pM[[4,2]]];
e6={1,1/2,-1,-1,-1,-1/2,1,1,1,1/2,-1,-1,-1/2,-1/4,1/2,1/2};
r7=Expand[pM[[4,3]]];
e7={1,-1/2,-1,1,-1,1/2,1,-1,1,-1/2,-1,1,-1/2,1/4,1/2,-1/2};
lY=mat2vec[Y];
MatrixForm[eM={e1,e2,e3,e4,e5,e6,e7}]
MatrixForm[ns=NullSpace[eM]]
nsM=Map[vec2m,ns];
Map[MatrixForm, nsM]
```

Now, let us describe the functions individually.

```
m=4;
mat2vec[m_]:=Flatten[m]
vec2m[v_]:={v[[1;;4]],v[[5;;8]],v[[9;;12]],v[[13;;16]]}
```

The transform is defined by a matrix cM (referred to as C in [1]).

$X = cM^T.Y.cM$ with $Y = eI \times Q$, thus Y are the de-quantized coefficients, while Q are the quantized coefficients.

```
MatrixForm[cM = {{1,1,1,1},{1,1/2, -1/2,
    -1},{1,-1,-1,1},{1/2,-1,1,-1/2}}]
```

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \frac{1}{2} & -\frac{1}{2} & -1 \\ 1 & -1 & -1 & 1 \\ \frac{1}{2} & -1 & 1 & -\frac{1}{2} \end{pmatrix}$$

```
MatrixForm[cT = Transpose[cM]]
```

$$\begin{pmatrix} 1 & 1 & 1 & \frac{1}{2} \\ 1 & \frac{1}{2} & -1 & -1 \\ 1 & -\frac{1}{2} & -1 & 1 \\ 1 & -1 & 1 & -\frac{1}{2} \end{pmatrix}$$

The quantized coefficients are multiplied with the following mask (we will, however, proceed with the discussion of the de-quantized coefficients). The solution for quantized coefficients can be straightforwardly derived by dividing the solutions of the de-quantized coefficients by the entries of eI.

```
MatrixForm[eI={{a^2, a*b, a^2, a*b}, {a*b, b^2, a*b, b^2}, {a^2, a*
    b, a^2, a*b}, {a*b, b^2,a*b, b^2}}]
```

$$\begin{pmatrix} a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \\ a^2 & ab & a^2 & ab \\ ab & b^2 & ab & b^2 \end{pmatrix}$$

```
a=1/2;   b=Sqrt[2/5];
```

```
MatrixForm[eI]
```

$$\begin{pmatrix} \frac{1}{4} & \frac{1}{\sqrt{10}} & \frac{1}{4} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{2}{5} & \frac{1}{\sqrt{10}} & \frac{2}{5} \\ \frac{1}{4} & \frac{1}{\sqrt{10}} & \frac{1}{4} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{2}{5} & \frac{1}{\sqrt{10}} & \frac{2}{5} \end{pmatrix}$$

We define the matrix Y, the de-quantized coefficients, in order to set up our system of linear equations.

```
MatrixForm[Y = {{y00, y01, y02, y03}, {y10, y11, y12, y13}, {y20,
    y21, y22, y23}, {y30, y31, y32, y33}}]
```

$$\begin{pmatrix} y00 & y01 & y02 & y03 \\ y10 & y11 & y12 & y13 \\ y20 & y21 & y22 & y23 \\ y30 & y31 & y32 & y33 \end{pmatrix}$$

The pixels pM are derived by Y.

```
pM = cT.(Y).cM;
```

The following equations define that the border pixels of the added noise pattern should be zero. After each line we write the coefficients of the above equation as vector.

```
r1 = Expand[pM[[1 ,4]]]
```

$$y00 - y01 + y02 - \frac{y03}{2} + y10 - y11 + y12 - \frac{y13}{2} + y20 - y21 + y22 - \frac{y23}{2} + \frac{y30}{2} - \frac{y31}{2} + \frac{y32}{2} - \frac{y33}{4}$$

We then set the weighting coefficient for each value within the vector:

```
e1={1,−1,1,−1/2,1,−1,1,−1/2,1,−1,1,−1/2,1/2,−1/2,1/2,−1/4};
```

The process is repeated for every border coefficients: (2,4), (3,4), (4,4)...

```
r2=Expand[pM[[2 ,4]]]
```

$$y00 - y01 + y02 - \frac{y03}{2} + \frac{y10}{2} - \frac{y11}{2} + \frac{y12}{2} - \frac{y13}{2} - y20 + y21 - y22 + \frac{y23}{2} - y30 + y31 - y32 + \frac{y33}{2}$$

```
e2={1,-1,1,-1/2,1/2,-1/2,1/2,-1/4,-1,1,-1,1/2,-1,1,-1,1/2};
```

```
r3=Expand[pM[[3 ,4]]]
```

$$y00 - y01 + y02 - \frac{y03}{2} - \frac{y10}{2} + \frac{y11}{2} - \frac{y12}{2} + \frac{y13}{4} - y20 + y21 - y22 + \frac{y23}{2} + y30 - y31 + y32 - \frac{y33}{2}$$

```
e3={1,-1,1,-1/2,-1/2,1/2,-1/2,1/4,-1,1,-1,1/2,1,-1,1,-1/2};
```

```
r4=Expand[pM[[4 ,4]]]
```

$$y00 - y01 + y02 - \frac{y03}{2} - y10 + y11 - y12 + \frac{y13}{2} + y20 - y21 + y22 - \frac{y23}{2} - \frac{y30}{2} + \frac{y31}{2} - \frac{y32}{2} + \frac{y33}{4}$$

```
e4={1,-1,1,-1/2,-1,1,-1,1/2,1,-1,1,-1/2,-1/2,1/2,-1/2,1/4};
```

```
r5=Expand[pM[[4 ,1]]]
```

$$y00 + y01 + y02 + \frac{y03}{2} - y10 - y11 - y12 - \frac{y13}{2} + y20 + y21 + y22 + \frac{y23}{2} - \frac{y30}{2} - \frac{y31}{2} - \frac{y32}{2} - \frac{y33}{4}$$

```
e5={1,1,1,1/2,-1,-1,-1,-1/2,1,1,1,1/2,-1/2,-1/2,-1/2,-1/4};
```

```
r6=Expand[pM[[4 ,2]]]
```

$$y00 + \frac{y01}{2} - y02 - y03 - y10 - \frac{y11}{2} + y12 + y13 + y20 + \frac{y21}{2} - y22 - y23 - \frac{y30}{2} - \frac{y31}{4} + \frac{y32}{2} + \frac{y33}{2}$$

```
e6={1,1/2,-1,-1,-1,-1/2,1,1,1,1/2,-1,-1,-1/2,-1/4,1/2,1/2};
```

```
r7=Expand[pM[[4 ,3]]]
```

$$y00-\frac{y01}{2}-y02+y03-y10+\frac{y11}{2}+y12-y13+y20-\frac{y21}{2}-y22+y23-\frac{y30}{2}+\frac{y31}{4}+\frac{y32}{2}-\frac{y33}{2}$$

```
e7={1,-1/2,-1,1,-1,1/2,1,-1,1,-1/2,-1,1,-1/2,1/4,1/2,-1/2};
```

```
lY=mat2vec[Y];
```

Finally, we write the equations as a matrix.

```
MatrixForm[eM={e1 , e2 , e3 , e4 , e5 , e6 , e7 }]
```

$$\begin{pmatrix}
1 & -1 & 1 & -\frac{1}{2} & 1 & -1 & 1 & -\frac{1}{2} & 1 & -1 & 1 & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{4} \\
1 & -1 & 1 & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{4} & -1 & 1 & -1 & \frac{1}{2} & -1 & 1 & -1 & \frac{1}{2} \\
1 & -1 & 1 & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{4} & -1 & 1 & -1 & \frac{1}{2} & 1 & -1 & 1 & -\frac{1}{2} \\
1 & -1 & 1 & -\frac{1}{2} & -1 & 1 & -1 & \frac{1}{2} & 1 & -1 & 1 & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{4} \\
1 & 1 & 1 & \frac{1}{2} & -1 & -1 & -1 & -\frac{1}{2} & 1 & 1 & 1 & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{4} \\
1 & \frac{1}{2} & -1 & -1 & -1 & -\frac{1}{2} & 1 & 1 & 1 & \frac{1}{2} & -1 & -1 & -\frac{1}{2} & -\frac{1}{4} & \frac{1}{2} & \frac{1}{2} \\
1 & -\frac{1}{2} & -1 & 1 & -1 & \frac{1}{2} & 1 & -1 & 1 & -\frac{1}{2} & -1 & 1 & -\frac{1}{2} & \frac{1}{4} & \frac{1}{2} & -\frac{1}{2}
\end{pmatrix}$$

Then we compute the basis of the NullSpace of eM, i.e., all vectors that are mapped to zero by eM.

```
MatrixForm[ns=NullSpace[eM]]
```

$$\begin{pmatrix}
\frac{1}{4} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 1 \\
-\frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
-\frac{1}{2} & 0 & 0 & -1 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0-1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & 0 & 0 & 1 & \frac{1}{2} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

There are 9 linear independent vectors, spanning the null space.

We can view these vectors also as matrices, that would be the noise patterns that can be added to the coefficients without changing the border pixels.

```
nsM=Map[vec2m, ns];
```

```
Map[MatrixForm, nsM]
```

$$\left\{\begin{pmatrix} \frac{1}{4} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -\frac{1}{2} & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2} & 0 & 0 & 1 \\ \frac{1}{2} & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -1 & 0 & 1 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}\right\}$$

Any combination of these patterns will keep the border coefficients unchanged. We have selected the following combination:

```
MatrixForm[s0=nsM[[8]]+nsM[[9]]]
```

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

As it omits the DC coefficient.

```
MatrixForm[s1=Transpose[s0]]
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Here, we check whether the two matrices are actually solutions .

```
eM.mat2vec[s1]
```

$$\{0, 0, 0, 0, 0, 0, 0\}$$

```
eM.mat2vec[s0]
```

$$\{0, 0, 0, 0, 0, 0, 0\}$$

References

[1] Iain E. G. Richardson, "H.264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia", Wiley, ISBN: 9780470848371, 2004.