

# The secret life of JPEG images

Forgery detection using compression traces

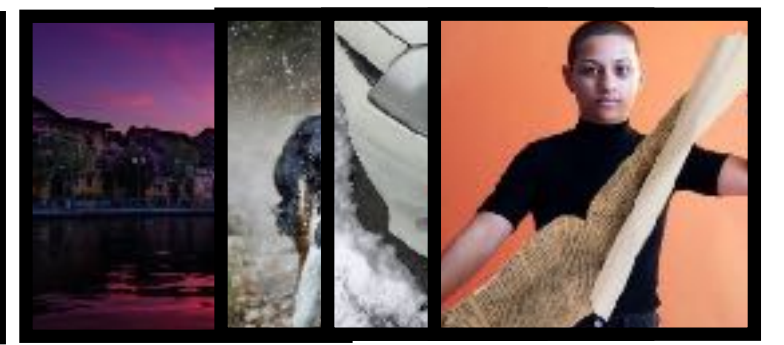
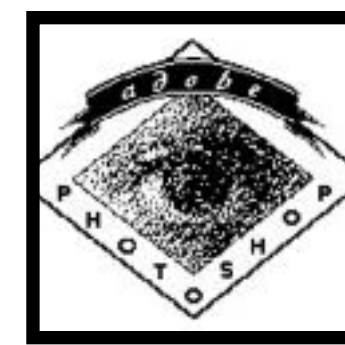
Tina Nikoukhah



école  
normale  
supérieure  
paris-saclay

université  
PARIS-SACLAY

# Image forgery



1840

1930  
1931

1968

1990

2000

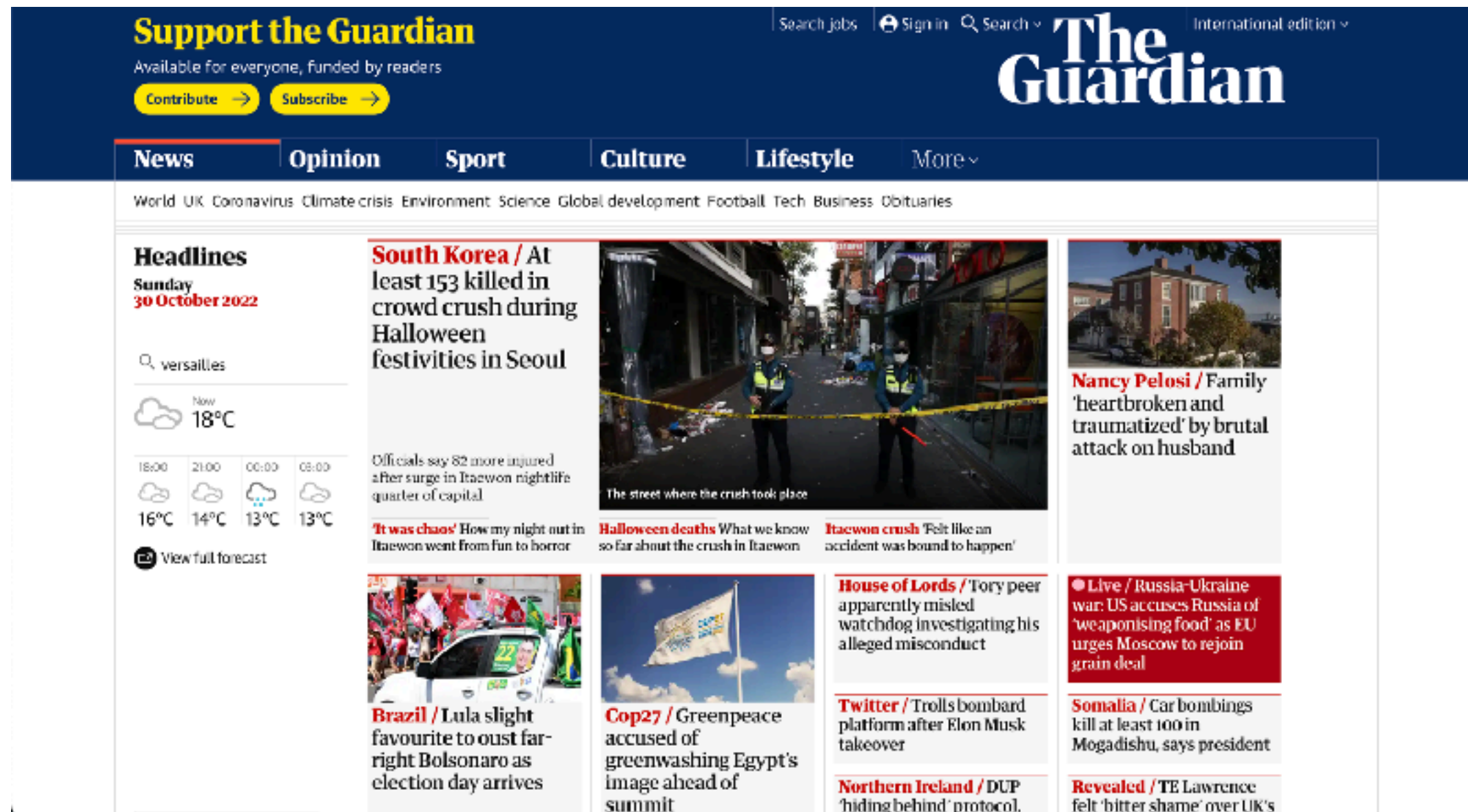
2010

2016

2017

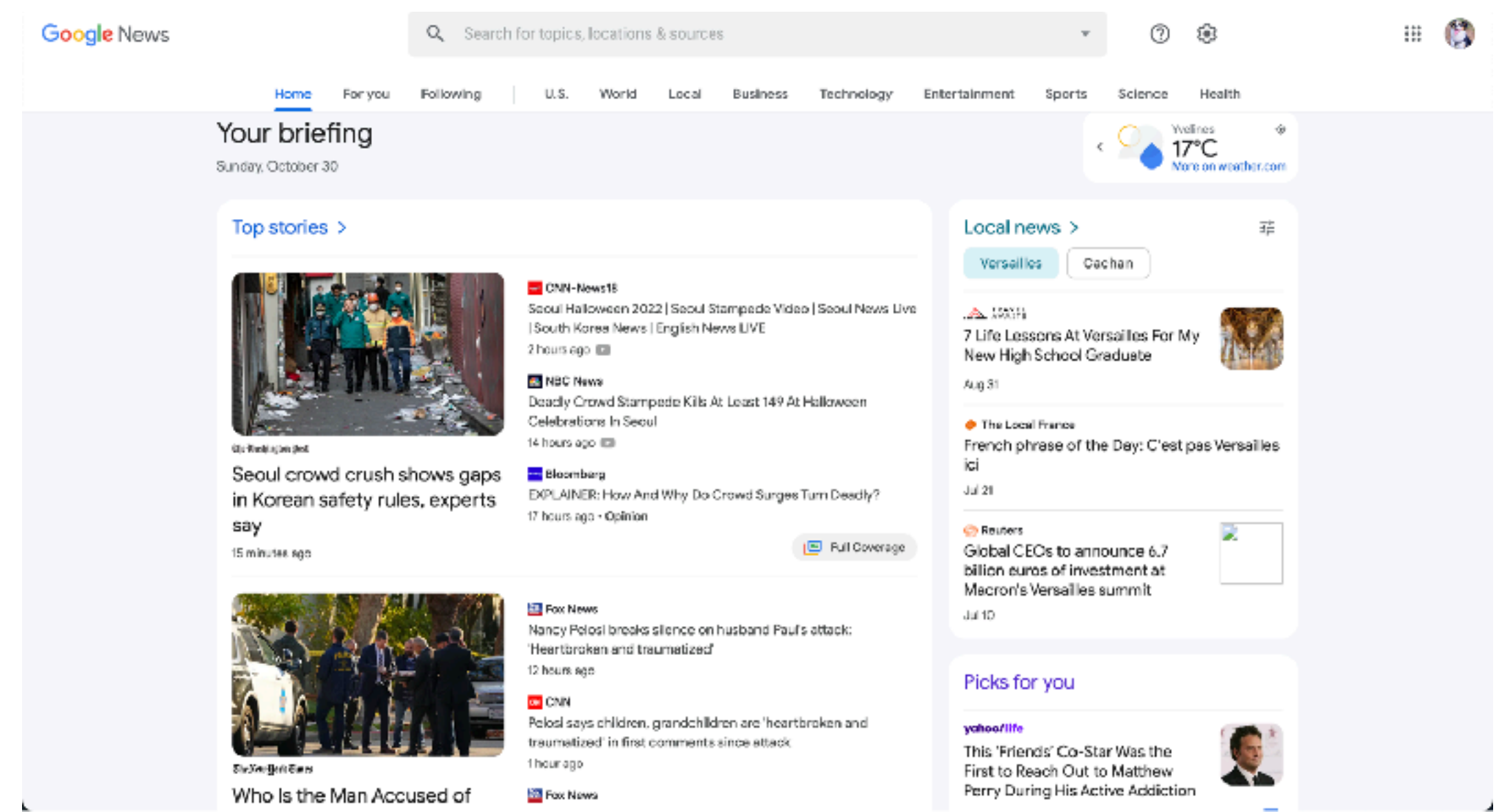
2018

# Images have become an important source of communication



The screenshot shows the homepage of The Guardian. At the top, there is a dark blue header with the text "Support the Guardian" and "Available for everyone, funded by readers". Below this, there are navigation links for "News", "Opinion", "Sport", "Culture", "Lifestyle", and "More". The main content area features a "Headlines" section for Sunday, 30 October 2022, with a search bar and a weather forecast for Versailles. The primary headline is "South Korea / At least 153 killed in crowd crush during Halloween festivities in Seoul", accompanied by a photograph of a street scene with police tape. Other headlines include "Nancy Pelosi / Family 'heartbroken and traumatized' by brutal attack on husband", "Brazil / Lula slight favourite to oust far-right Bolsonaro as election day arrives", "Cop27 / Greenpeace accused of greenwashing Egypt's image ahead of summit", "House of Lords / Tory peer apparently misled watchdog investigating his alleged misconduct", "Twitter / Trolls bombard platform after Elon Musk takeover", "Northern Ireland / DUP 'hiding behind' protocol", "Somalia / Car bombings kill at least 100 in Mogadishu, says president", and "Revealed / TE Lawrence felt 'bitter shame' over UK's".

theguardian



The screenshot shows the Google News homepage. At the top, there is a search bar and navigation links for "Home", "For you", "Following", "U.S.", "World", "Local", "Business", "Technology", "Entertainment", "Sports", "Science", and "Health". The main content area features a "Your briefing" section for Sunday, October 30, with a weather forecast for Versailles. The primary headline is "Seoul crowd crush shows gaps in Korean safety rules, experts say", accompanied by a photograph of a street scene with police tape. Other headlines include "Nancy Pelosi breaks silence on husband Paul's attack: 'Heartbroken and traumatized'", "Polosol says children, grandchildren are 'heartbroken and traumatized' in first comments since attack", "7 Life Lessons At Versailles For My New High School Graduate", "French phrase of the Day: C'est pas Versailles ici", "Global CEOs to announce 6.7 billion euros of investment at Macron's Versailles summit", and "This 'Friends' Co-Star Was the First to Reach Out to Matthew Perry During His Active Addiction".

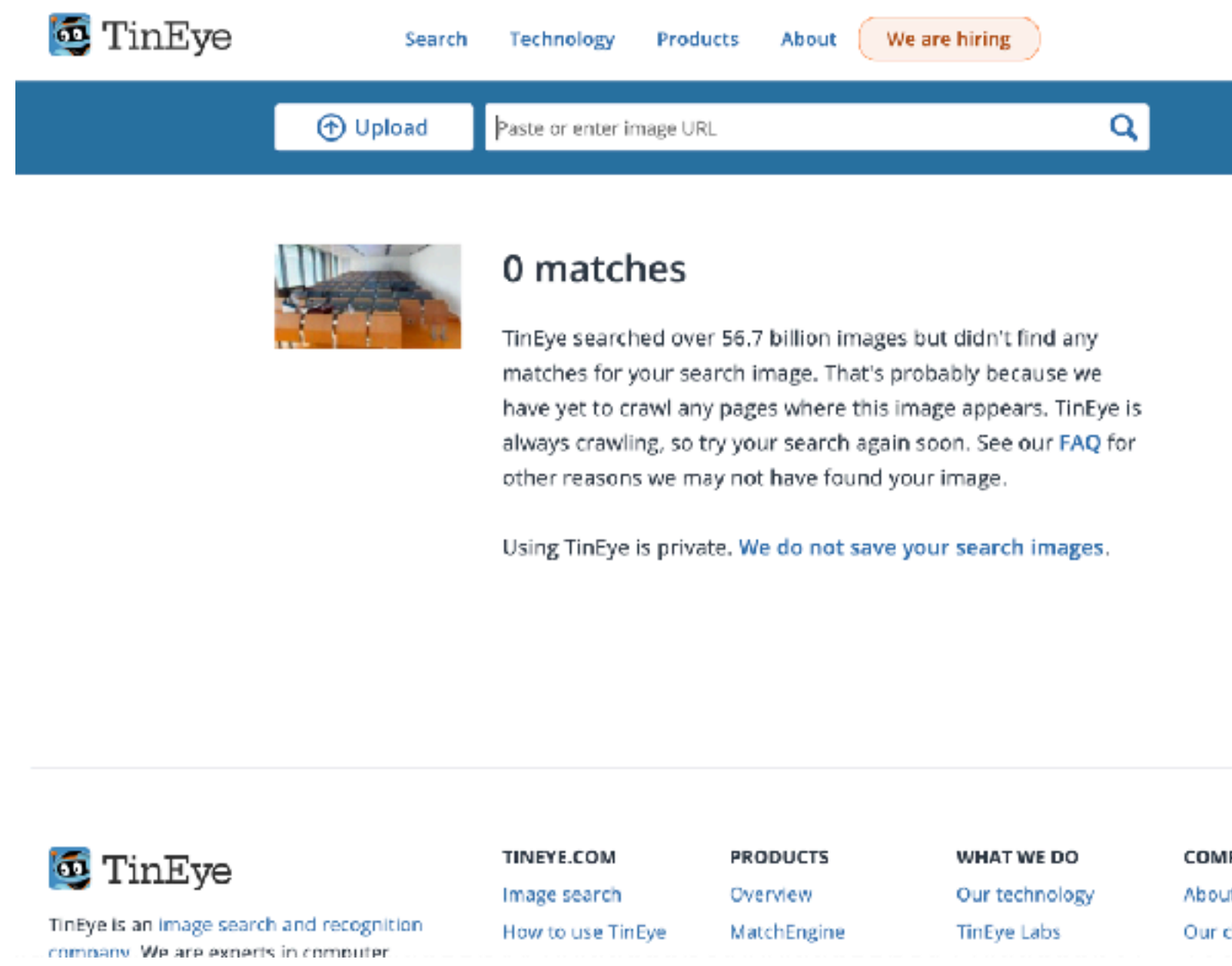
Google News

# A picture is worth a thousand words

Twitter search results for #TinasPhdDefense. The page shows two tweets from user @Pashm\_ina. The first tweet includes a photo of a large, empty lecture hall with many rows of orange chairs and blue desks. The second tweet includes a similar photo showing the same lecture hall, but with only one person visible in the distance. The right sidebar shows trending topics like 'Fake news' and 'China', and a 'Who to follow' section with users like Planar Polarity Team, MAMMOth, and Pantelis Dogoulis.

# First reflexes to have

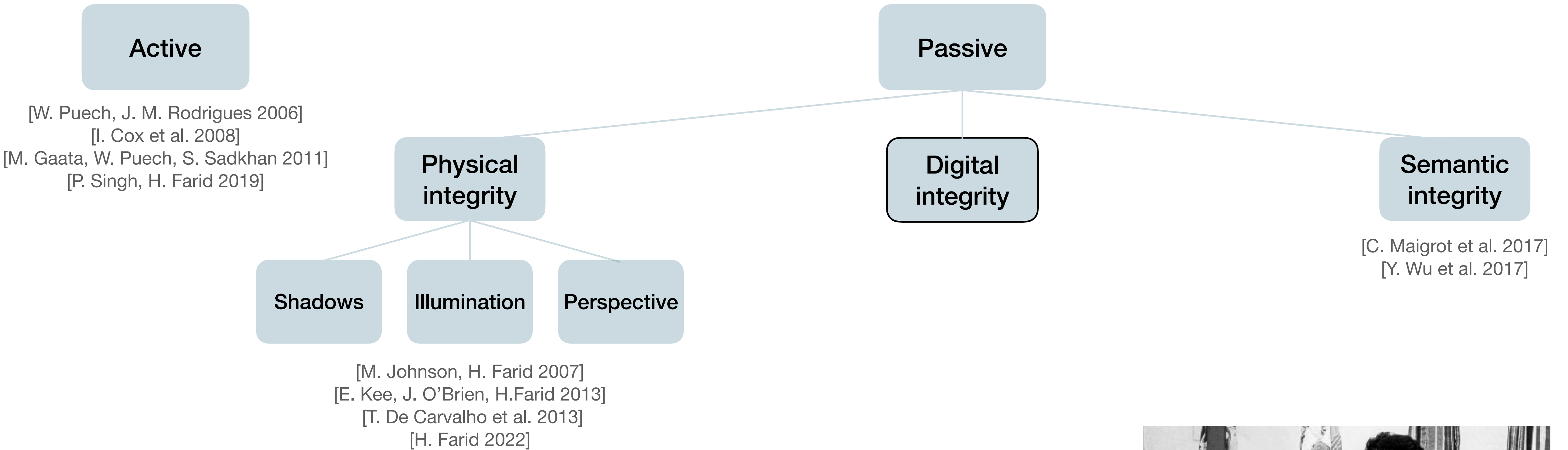
- Look at the image.
- Analyse the image file, but there are no EXIF metadata!
- Perform image reverse search on the Internet, to identify the image's source.



Fake information

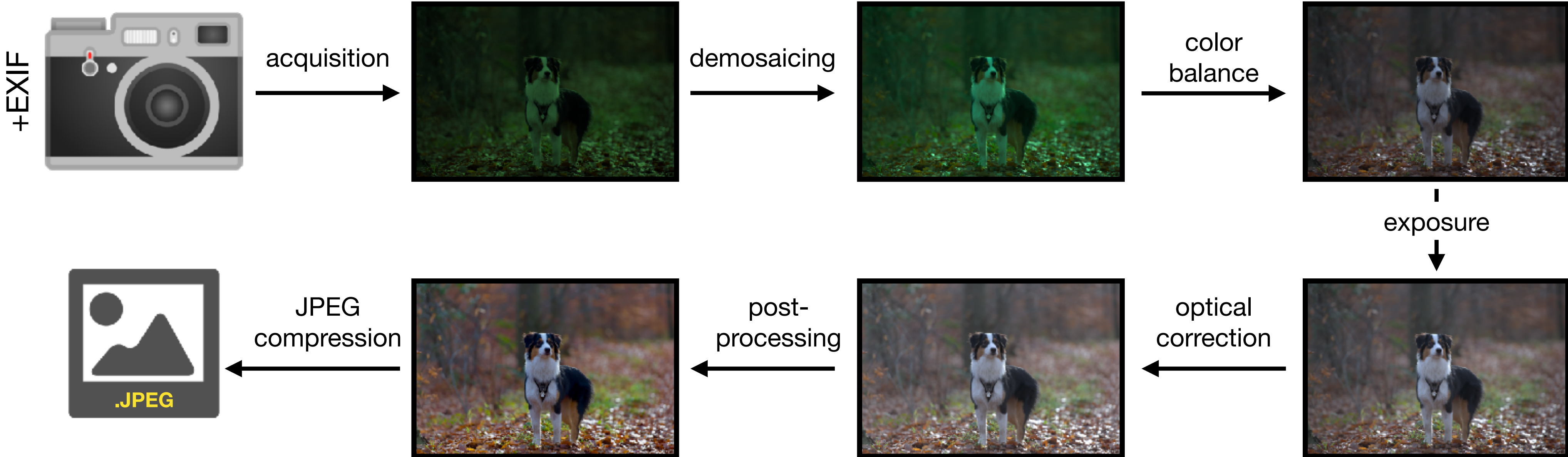
**How to verify an image just with the image itself?**

# An overview on image forensics



Kim Jong-Un with Elvis. Photoshop: Lazaro Gamio/Axios

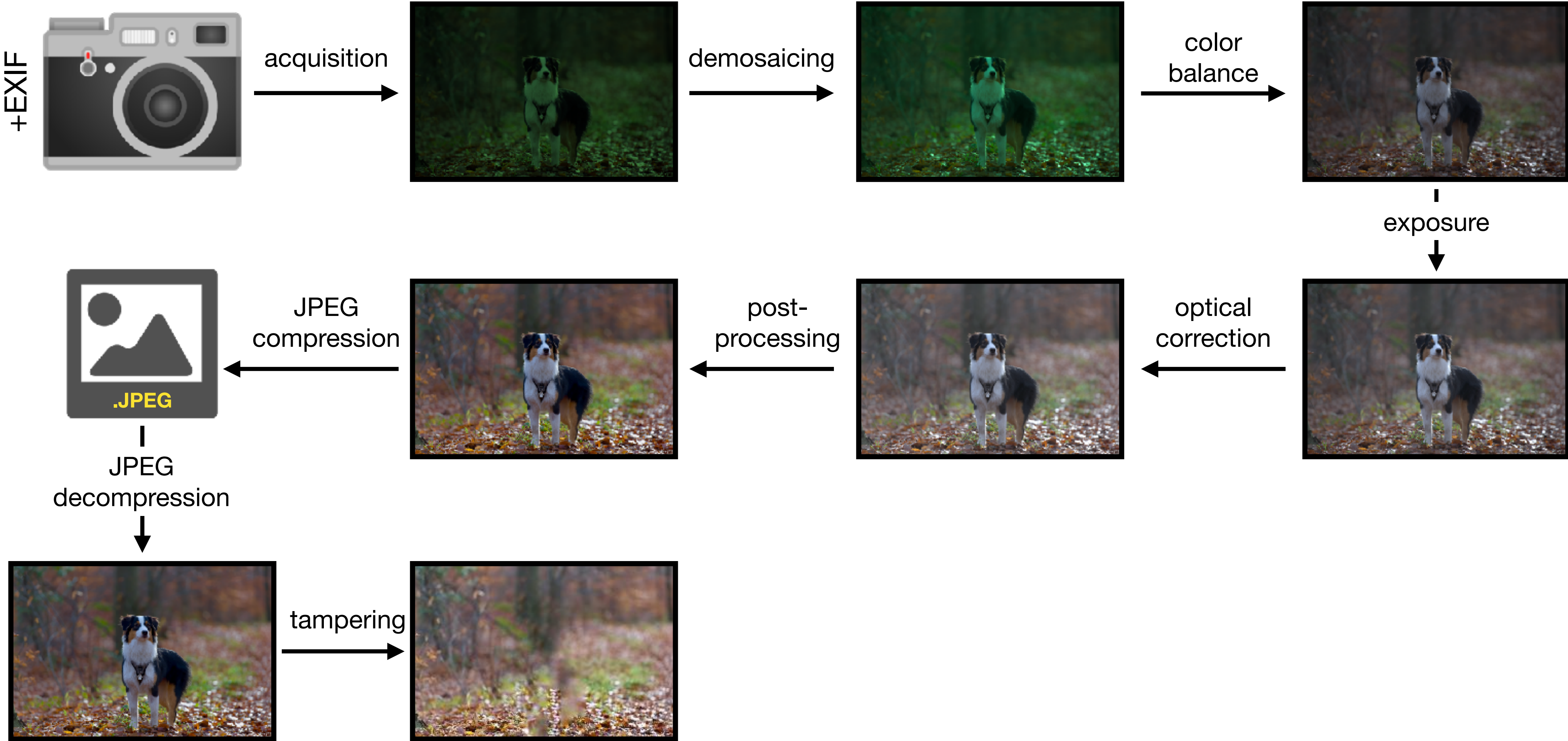
# Image processing pipeline



Simplified processing pipeline of an image.

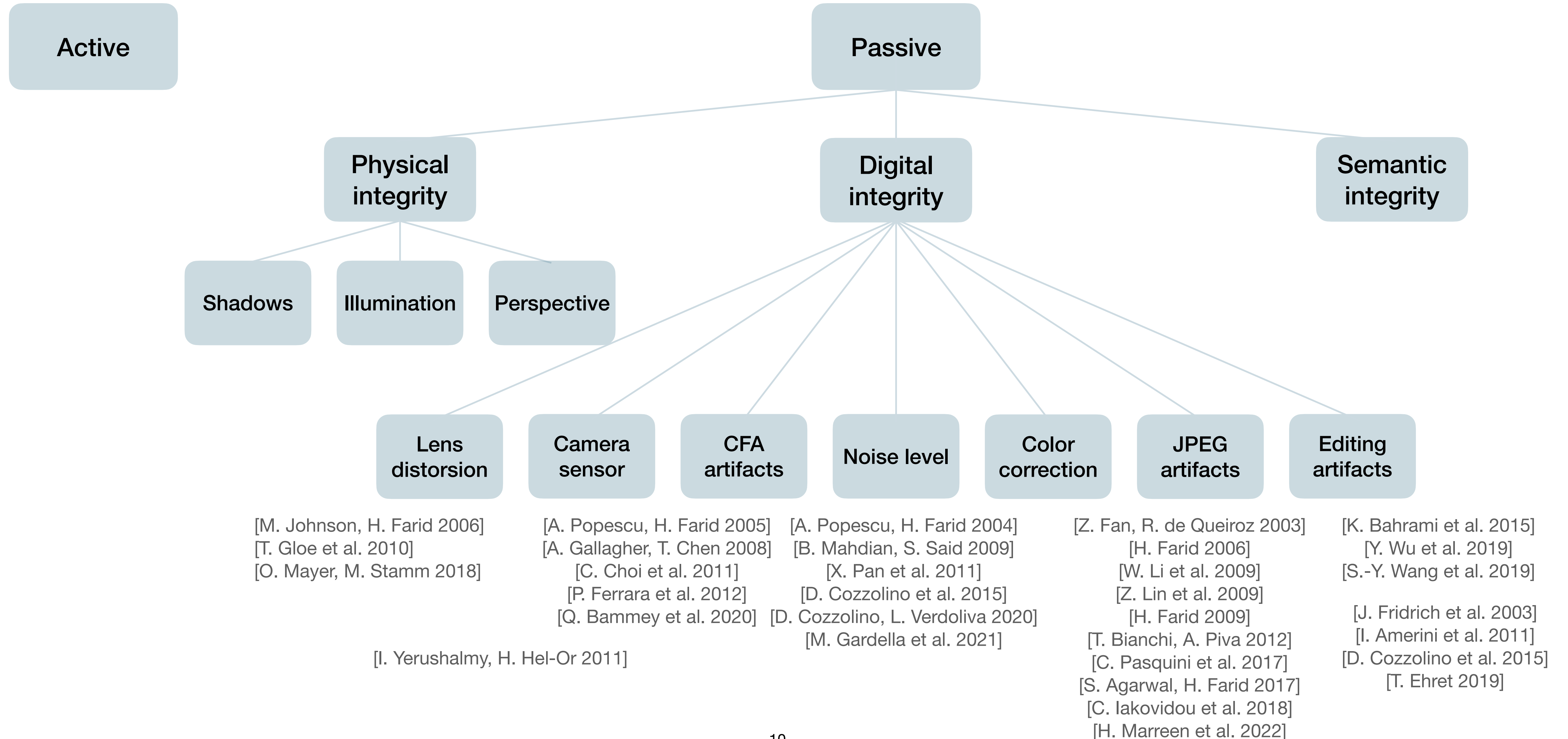


# Image processing pipeline



All these operations provide important clues that are exploited in the image forensic analysis.

# An overview on image forensics



# Forgery: *inpainting*

ENS Paris-Saclay  
@ENS\_ParisSaclay

Rehearsal of [#TinasPhdDefense](#), sound check, video check, internet check 🙄🙄🙄



10:49 AM · Nov 1, 2022 · Twitter Web App

Original image.

Alicia Keys  
@aliciakeys

No one, no one, no one 🎵 [#TinasPhdDefense](#)



4:02 PM · Nov 8, 2022 ·

Fake post: parts of the image have been erased.

# Forgery: *copy-move*

 **ENS Paris-Saclay**  
@ENS\_ParisSaclay

Rehearsal of [#TinasPhdDefense](#), sound check, video check, internet check 🙄🙄🙄



10:49 AM · Nov 1, 2022 · Twitter Web App

Original image.

 **Movers**  
@troll

Soooo many people at [#TinasPhdDefense](#) 🤡🤡🤡



4:02 PM · Nov 8, 2022 · Twitter App

Fake post: parts of the image have been duplicated.

# Forgery: *splicing*

ENS Paris-Saclay  
@ENS\_ParisSaclay

Rehearsal of [#TinasPhdDefense](#), sound check, video check, internet check 🙄🙄🙄



10:49 AM · Nov 1, 2022 · Twitter Web App

Original image.

Pashmina fan club  
@Pashm\_ina

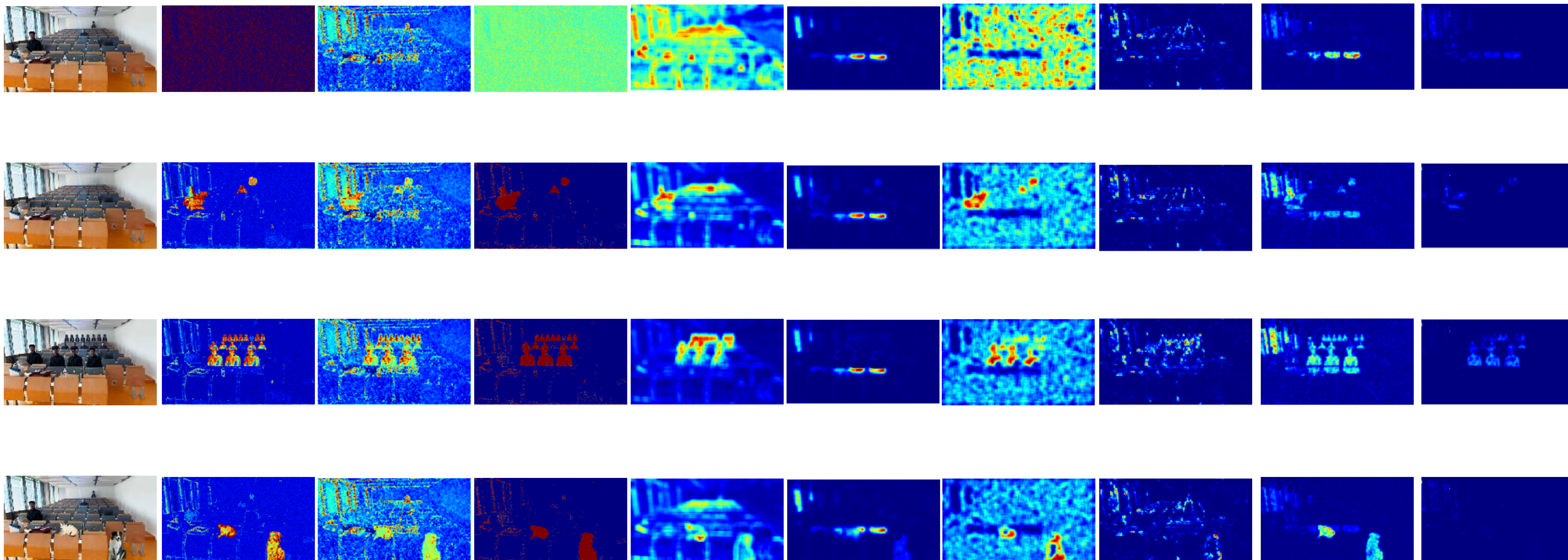
Her pets came to [#TinasPhdDefense](#) !! 🐱🐶



4:05 PM · Nov 8, 2022 · Twitter Web App

Fake post: new objects have been inserted in the image.

# Passive image forgery detection methods



input

DCT

[Ye et al. 2007]

BLK

[Li et al. 2009]

DQ

[Lin et al. 2009]

Ghost

[Farid 2009]

Splicebuster

[Cozzolino  
et al. 2015]

CAGI

[Iakovidou  
et al. 2018]

ManTraNet

[Wu et al. 2019]

Noiseprint

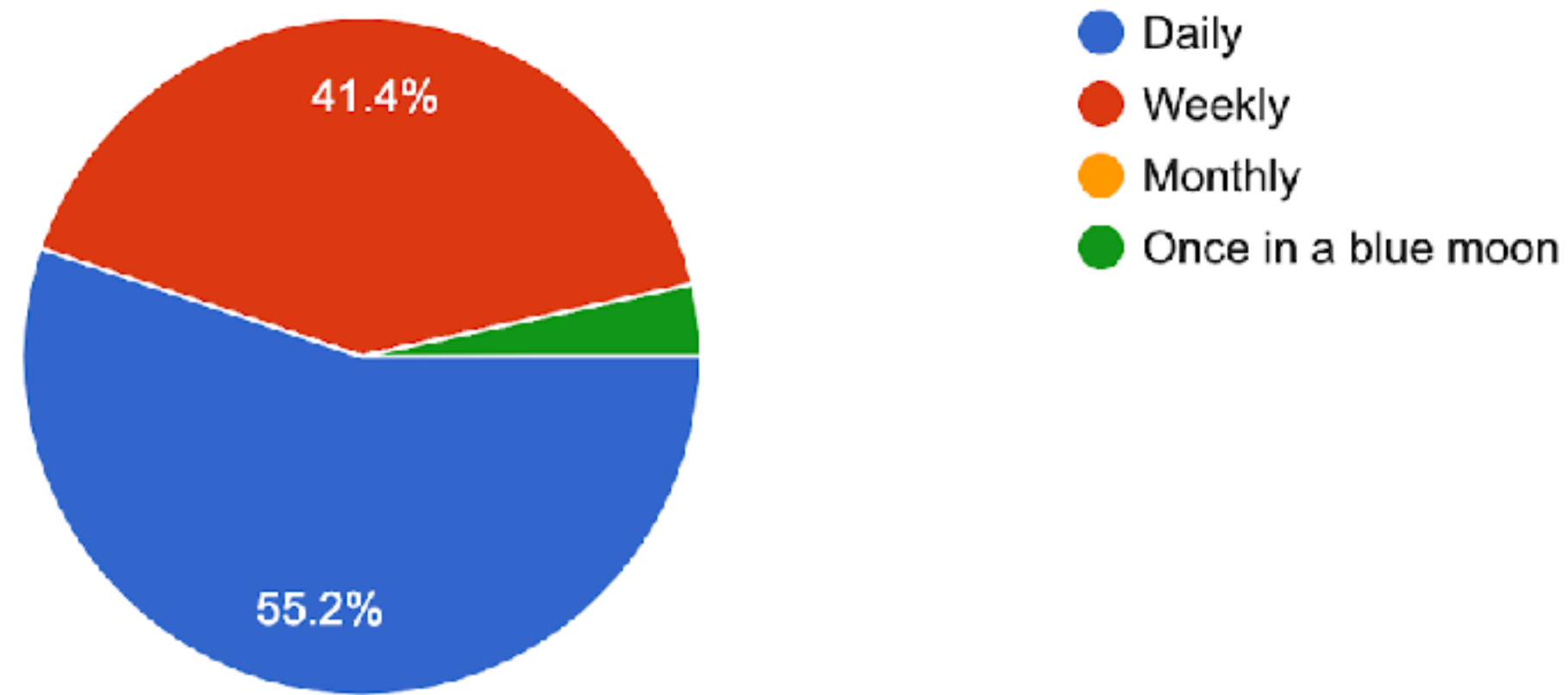
[Cozzolino  
et al. 2020]

Comprint

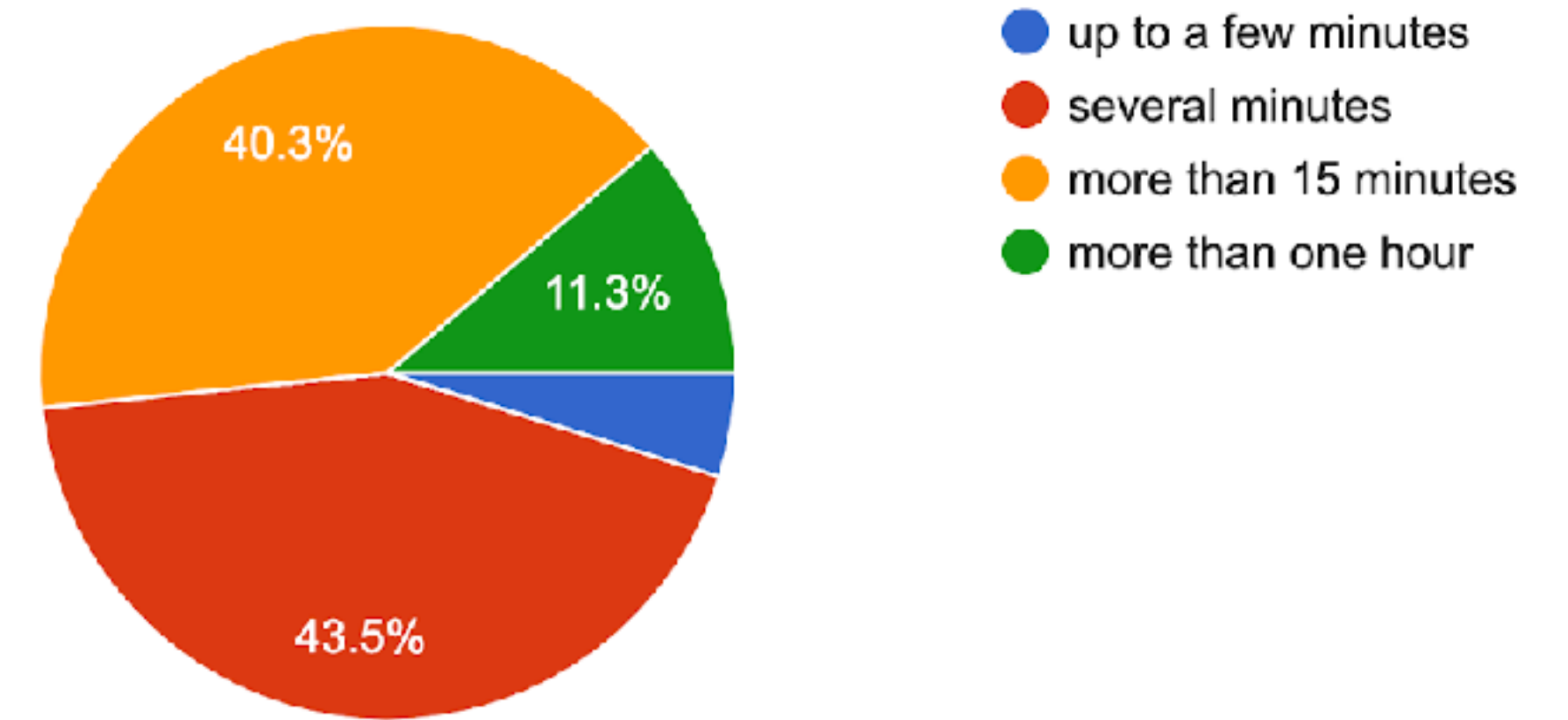
[Mareen  
et al. 2022]

# Short survey among AFP fact-checkers during the Envisu4 project

How often do you need to verify an image with respect to its authenticity ?  
62 responses



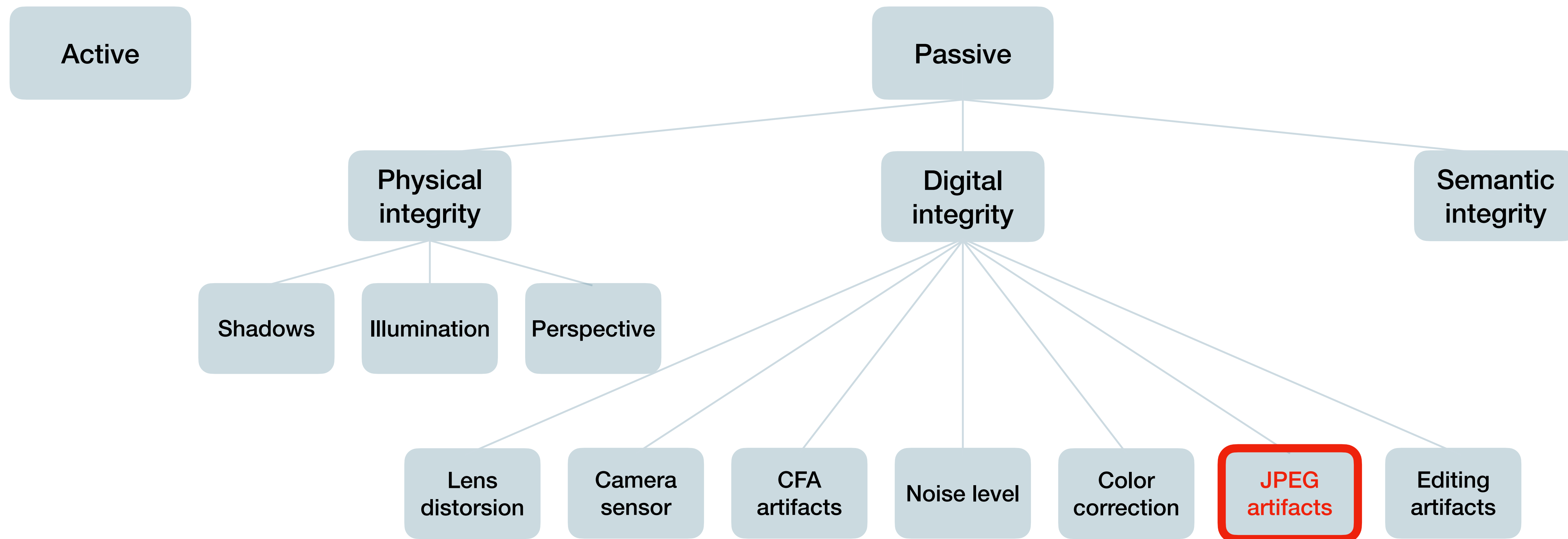
How much time do you typically spend on the verification of an image ?  
62 responses



- The tools are “interesting” and “attractive”, but they are also “difficult to use”.
- They wish “more explanation”, “less false positives” and “interpretability”.

*"It's a bit hard to understand and therefore not as useful as it could be." - AFP journalist*

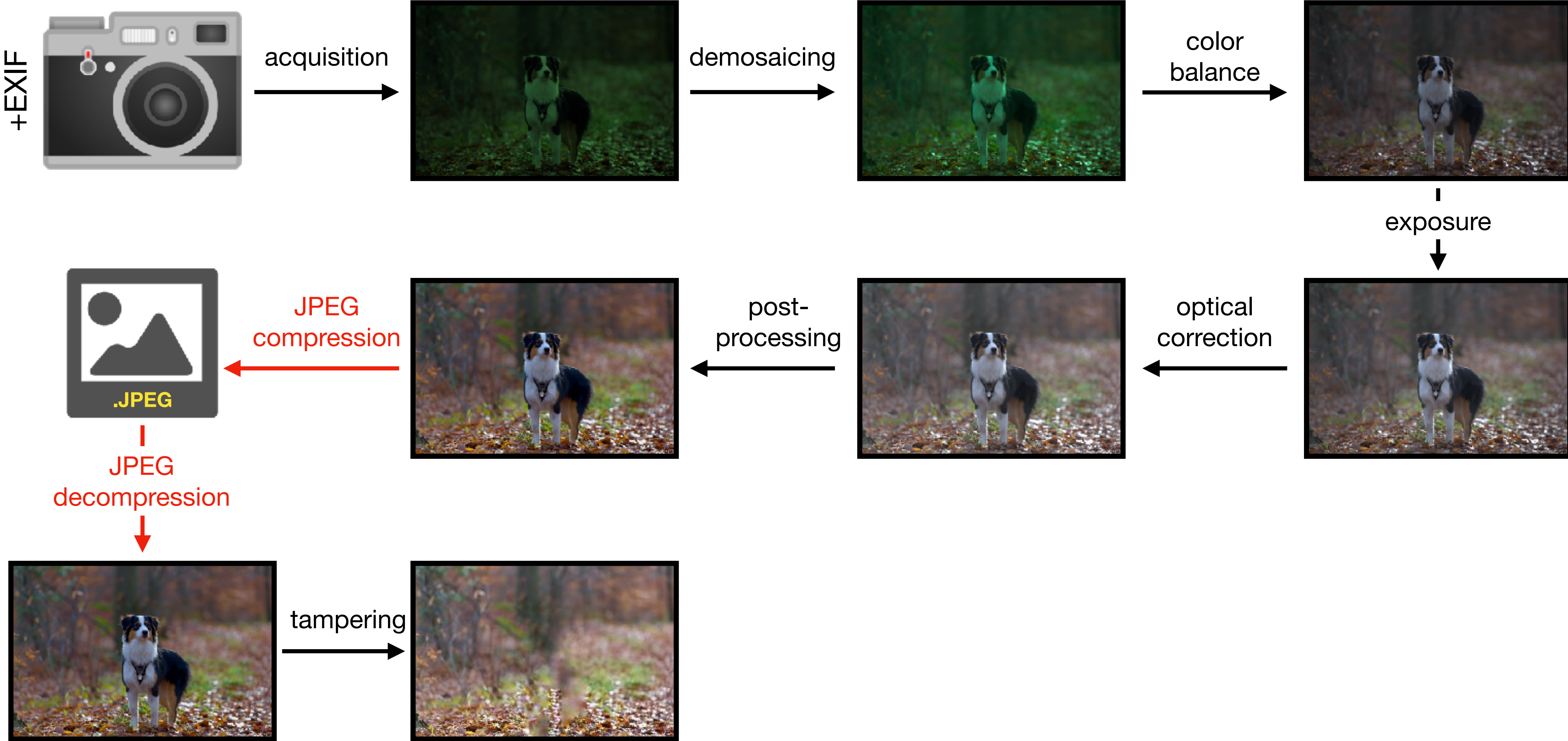
# Where do we stand?



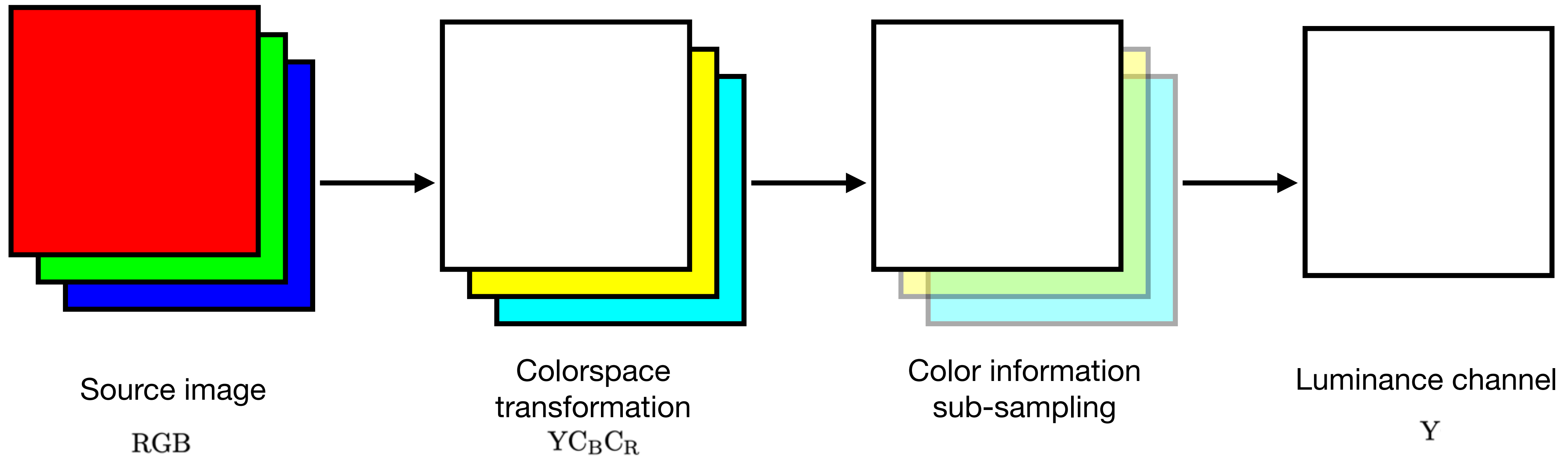
Automatic task-based algorithms with controlled false alarms.



# Image processing pipeline

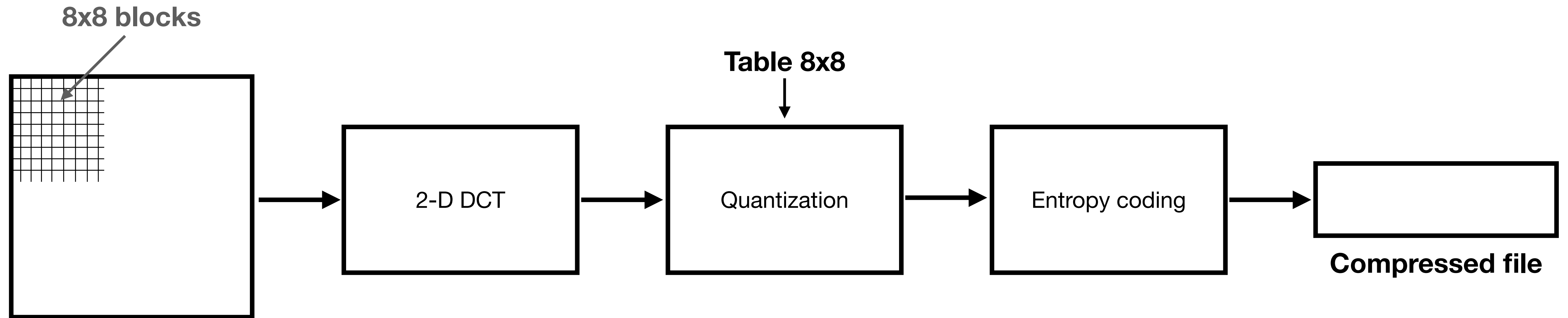


# JPEG compression pipeline



$$\begin{pmatrix} Y \\ C_B \\ C_R \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.5 \\ 0.5 & -0.419 & 0.081 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix}$$

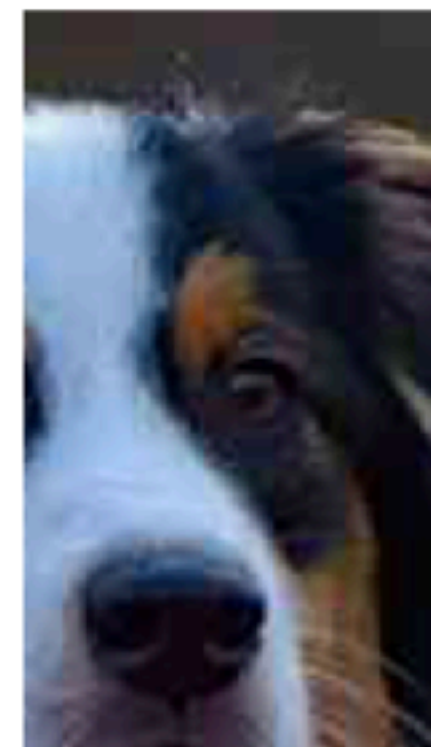
# JPEG compression pipeline



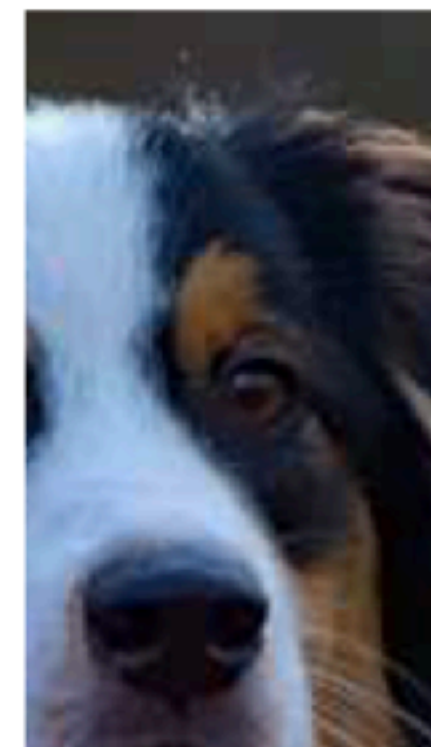
Quality factor  
QF = {1...100}



QF = 10  
398 Ko



QF = 30  
793 Ko



QF = 50  
1.2 Mo

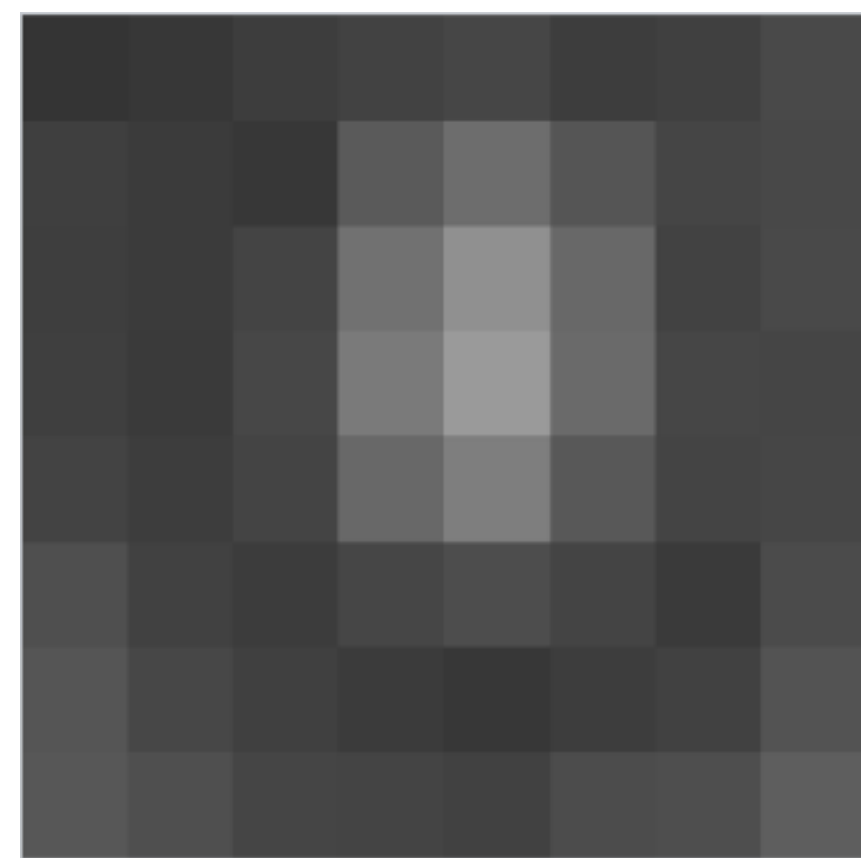
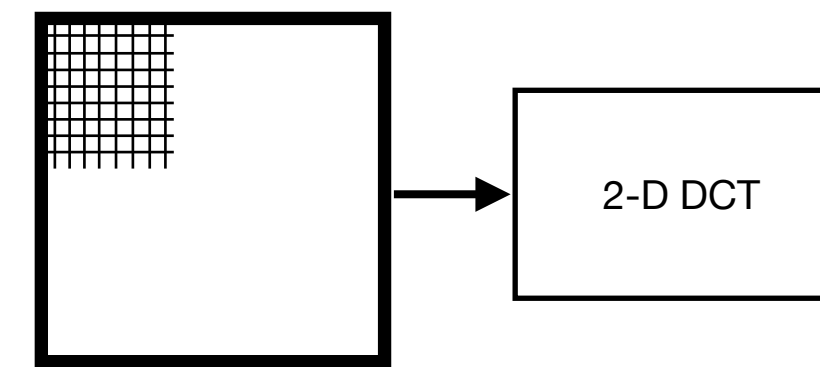


QF = 70  
2.0 Mo



QF = 90  
5.3 Mo

# JPEG compression pipeline



8x8 block

-76	-73	-67	-62	-58	-67	-64	-55
-65	-69	-73	-38	-19	-43	-59	-56
-66	-69	-60	-15	16	-24	-62	-55
-65	-70	-57	-6	26	-22	-58	-59
-61	-67	-60	-24	-2	-40	-60	-58
-49	-63	-68	-58	-51	-60	-70	-53
-43	-57	-64	-69	-73	-67	-63	-45
-41	-49	-59	-60	-63	-52	-50	-34

Pixel values - 128:  $Y$



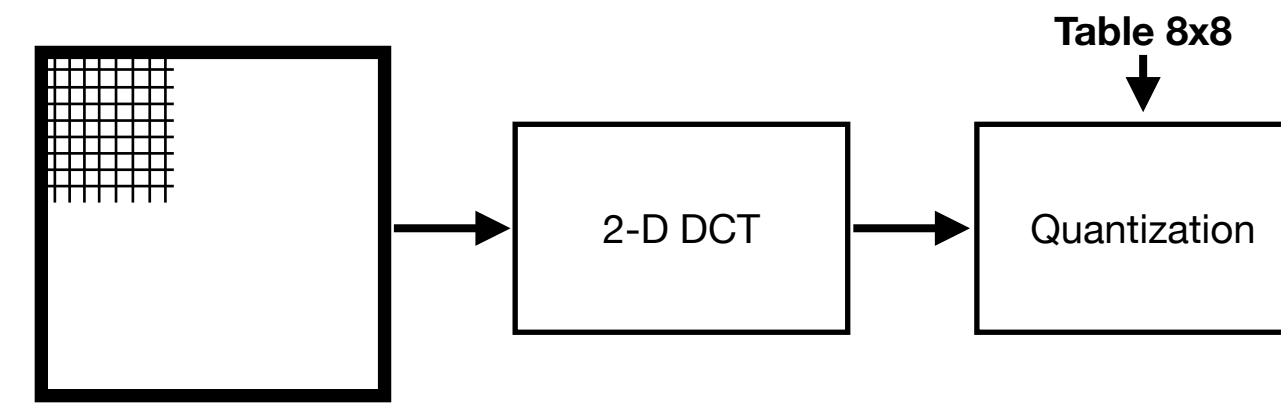
-415.4	-30.19	-61.20	27.24	56.12	-20.10	-2.39	0.46
4.47	-21.86	-60.76	10.25	13.15	-7.09	-8.54	4.88
-46.83	7.37	77.13	-24.56	-28.91	9.93	5.42	-5.65
-48.53	12.07	34.10	-14.76	-10.24	6.30	1.83	1.95
12.12	-6.55	-13.20	-3.95	-1.87	1.75	-2.79	3.14
-7.73	2.91	2.38	-5.94	-2.38	0.94	4.30	1.85
-1.03	0.18	0.42	-2.42	-0.88	-3.02	4.12	-0.66
-0.17	0.14	-1.07	-4.19	-1.17	-0.10	0.50	1.68

DCT coefficients:  $I$

$$I(u, v) = \frac{1}{4} T_u T_v \sum_{i=0}^7 \sum_{j=0}^7 Y(i, j) \cos\left(\frac{(2i+1)u\pi}{16}\right) \cos\left(\frac{(2j+1)v\pi}{16}\right),$$

where  $T_u = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } u = 0. \\ 1 & \text{for } u > 0. \end{cases}$   $u$  is the horizontal spatial frequency  
 $v$  is the vertical spatial frequency

# JPEG compression pipeline

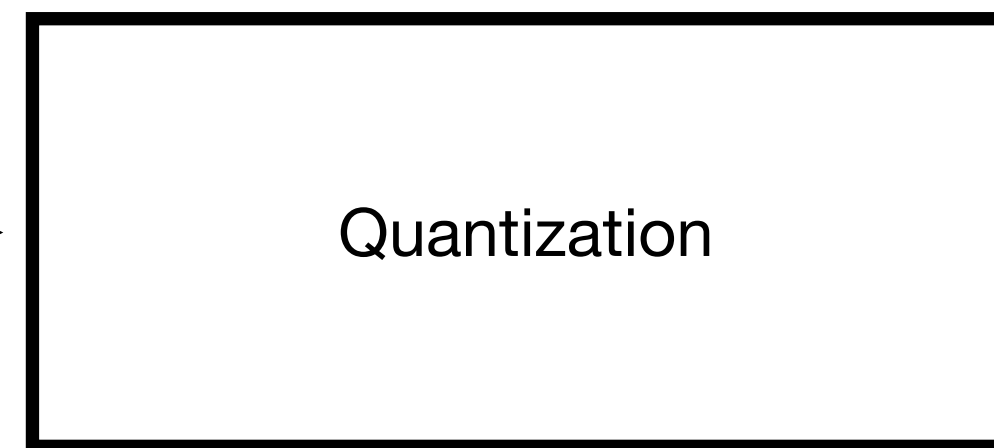


Quantization table:  $Q$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

-415.4	-30.19	-61.20	27.24	56.12	-20.10	-2.39	0.46
4.47	-21.86	-60.76	10.25	13.15	-7.09	-8.54	4.88
-46.83	7.37	77.13	-24.56	-28.91	9.93	5.42	-5.65
-48.53	12.07	34.10	-14.76	-10.24	6.30	1.83	1.95
12.12	-6.55	-13.20	-3.95	-1.87	1.75	-2.79	3.14
-7.73	2.91	2.38	-5.94	-2.38	0.94	4.30	1.85
-1.03	0.18	0.42	-2.42	-0.88	-3.02	4.12	-0.66
-0.17	0.14	-1.07	-4.19	-1.17	-0.10	0.50	1.68

DCT coefficients:  $I$

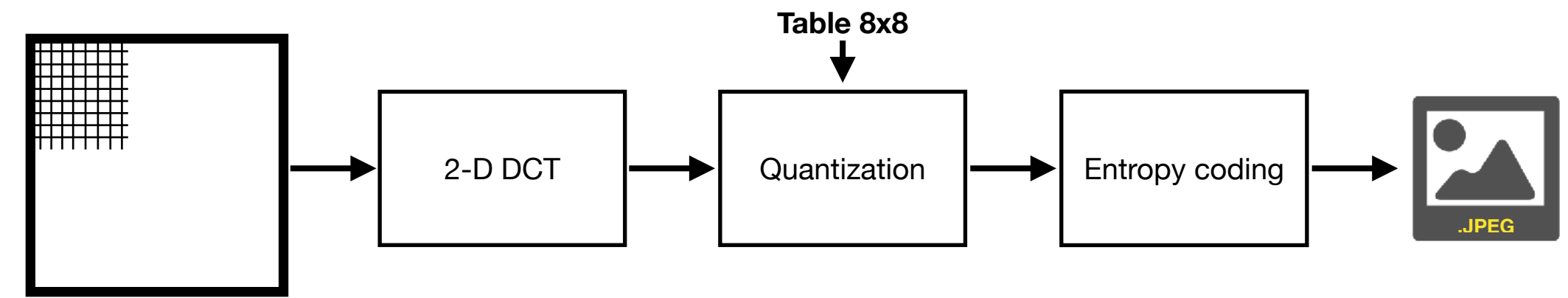


$$D(u, v) = \text{round} \left( \frac{I(u, v)}{Q(u, v)} \right)$$

-26	-3	-6	2	2	-1	0	0
0	-2	-4	1	1	0	0	0
-3	1	5	-1	-1	0	0	0
-3	1	2	-1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Quantized DCT coefficients:  $D$

# JPEG compression pipeline

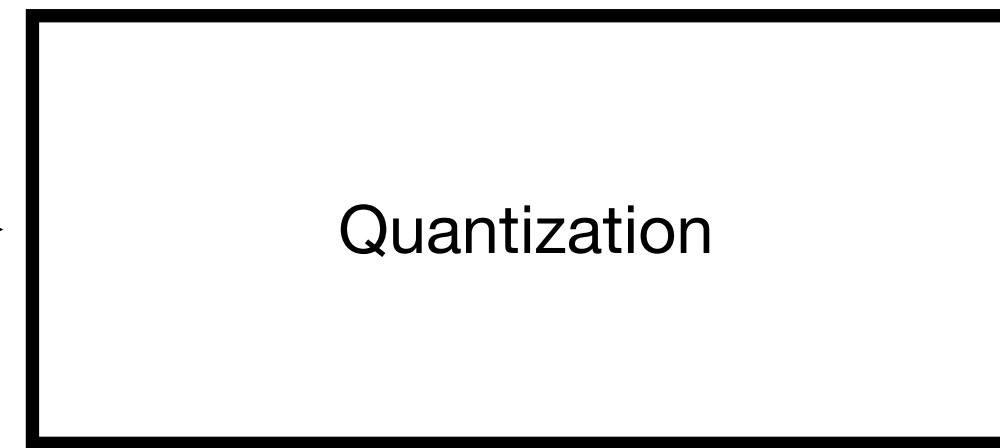


Quantization table:  $Q$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

-415.4	-30.19	-61.20	27.24	56.12	-20.10	-2.39	0.46
4.47	-21.86	-60.76	10.25	13.15	-7.09	-8.54	4.88
-46.83	7.37	77.13	-24.56	-28.91	9.93	5.42	-5.65
-48.53	12.07	34.10	-14.76	-10.24	6.30	1.83	1.95
12.12	-6.55	-13.20	-3.95	-1.87	1.75	-2.79	3.14
-7.73	2.91	2.38	-5.94	-2.38	0.94	4.30	1.85
-1.03	0.18	0.42	-2.42	-0.88	-3.02	4.12	-0.66
-0.17	0.14	-1.07	-4.19	-1.17	-0.10	0.50	1.68

DCT coefficients:  $I$

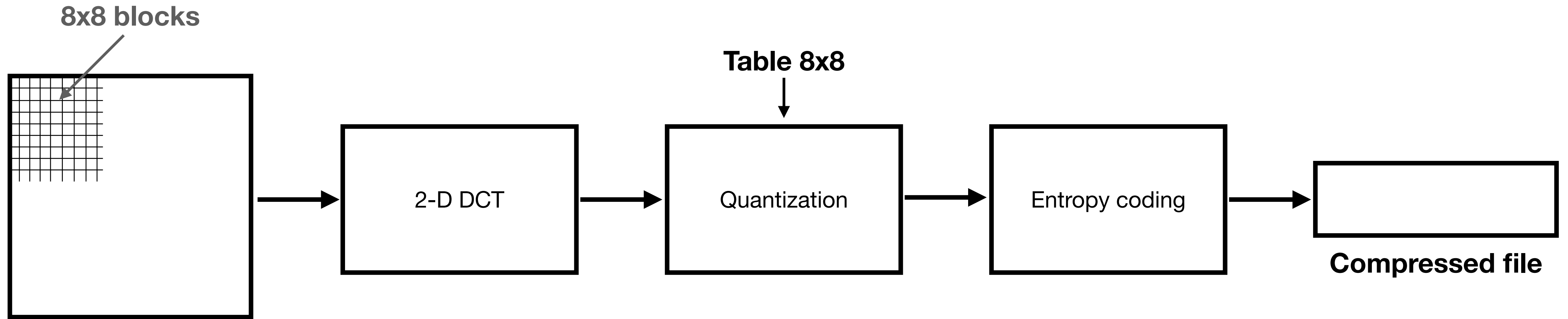


$$D(u, v) = \text{round} \left( \frac{I(u, v)}{Q(u, v)} \right)$$

-26	-3	-6	2	2	-1	0	0
0	-2	-4	1	1	0	0	0
-8	1	5	-1	-1	0	0	0
-3	1	2	-1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Quantized DCT coefficients:  $D$

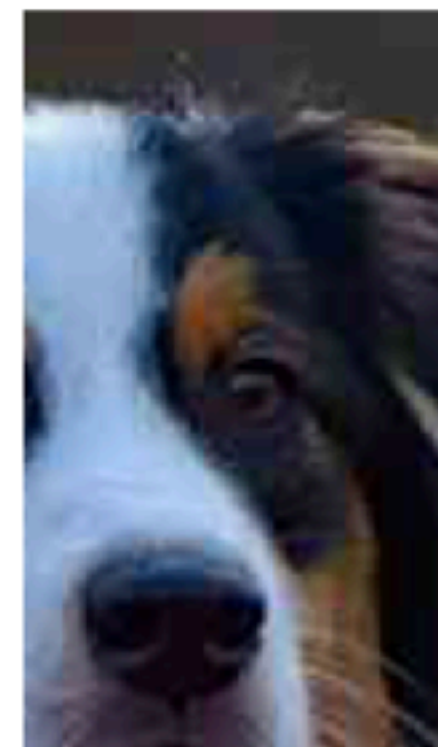
# JPEG compression pipeline



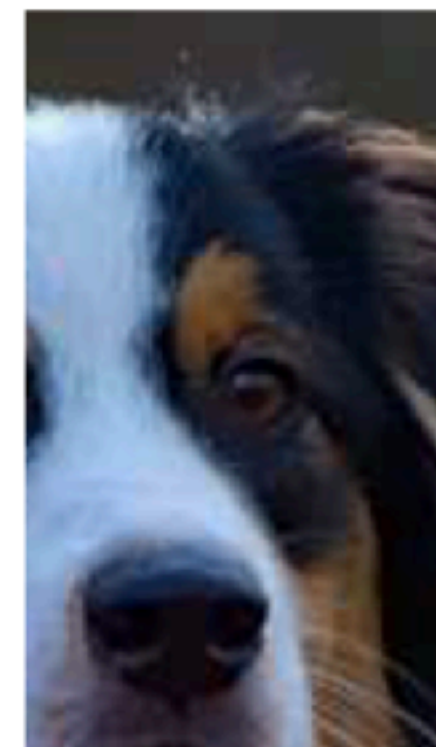
Quality factor  
QF = {1...100}



QF = 10  
398 Ko



QF = 30  
793 Ko



QF = 50  
1.2 Mo

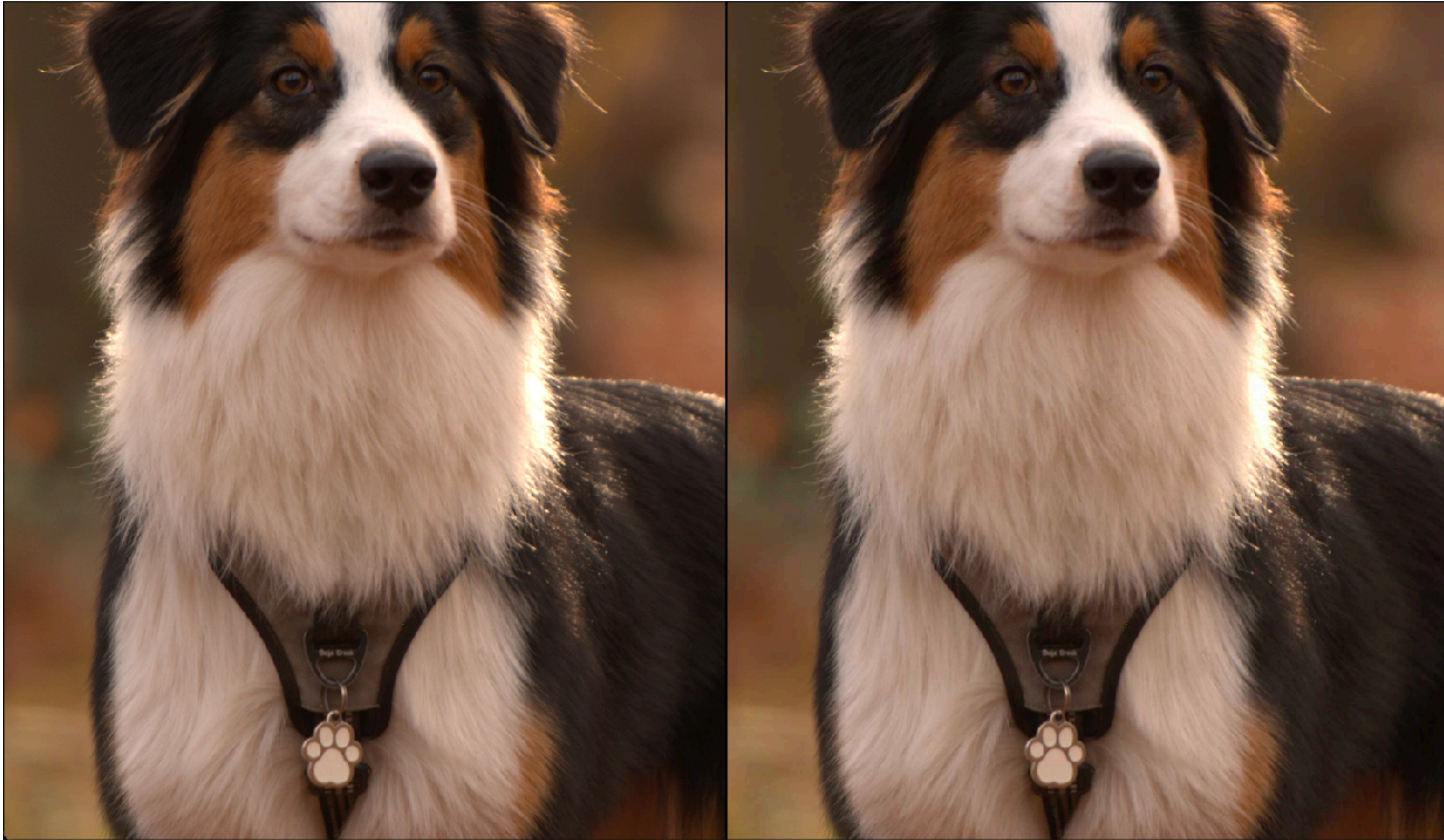


QF = 70  
2.0 Mo



QF = 90  
5.3 Mo

# JPEG grid artifact



Uncompressed image

Compressed image at QF = 50



# JPEG grid artifact



Uncompressed image

Compressed image at QF = 50

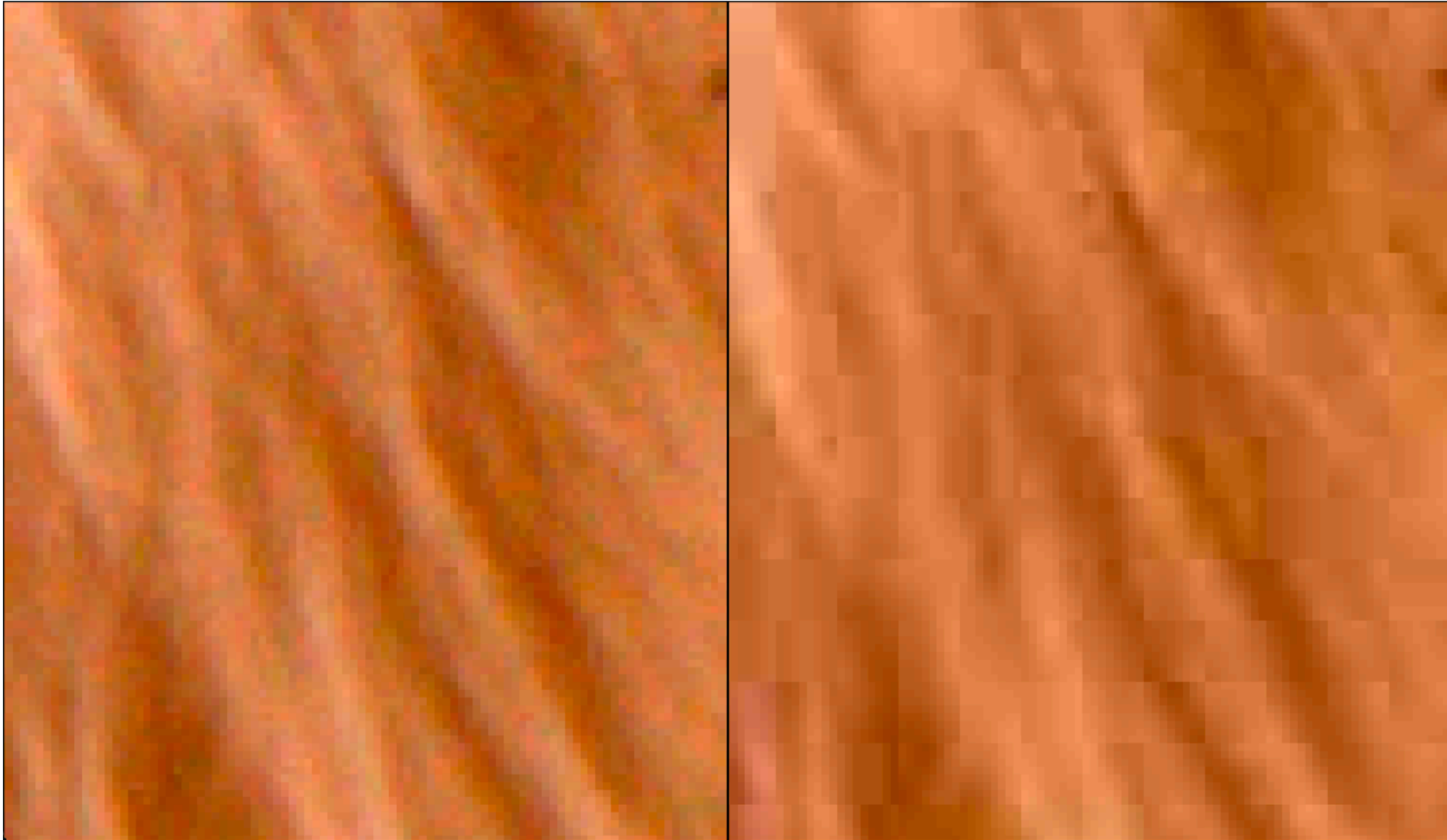
# JPEG grid artifact



Uncompressed image

Compressed image at QF = 50

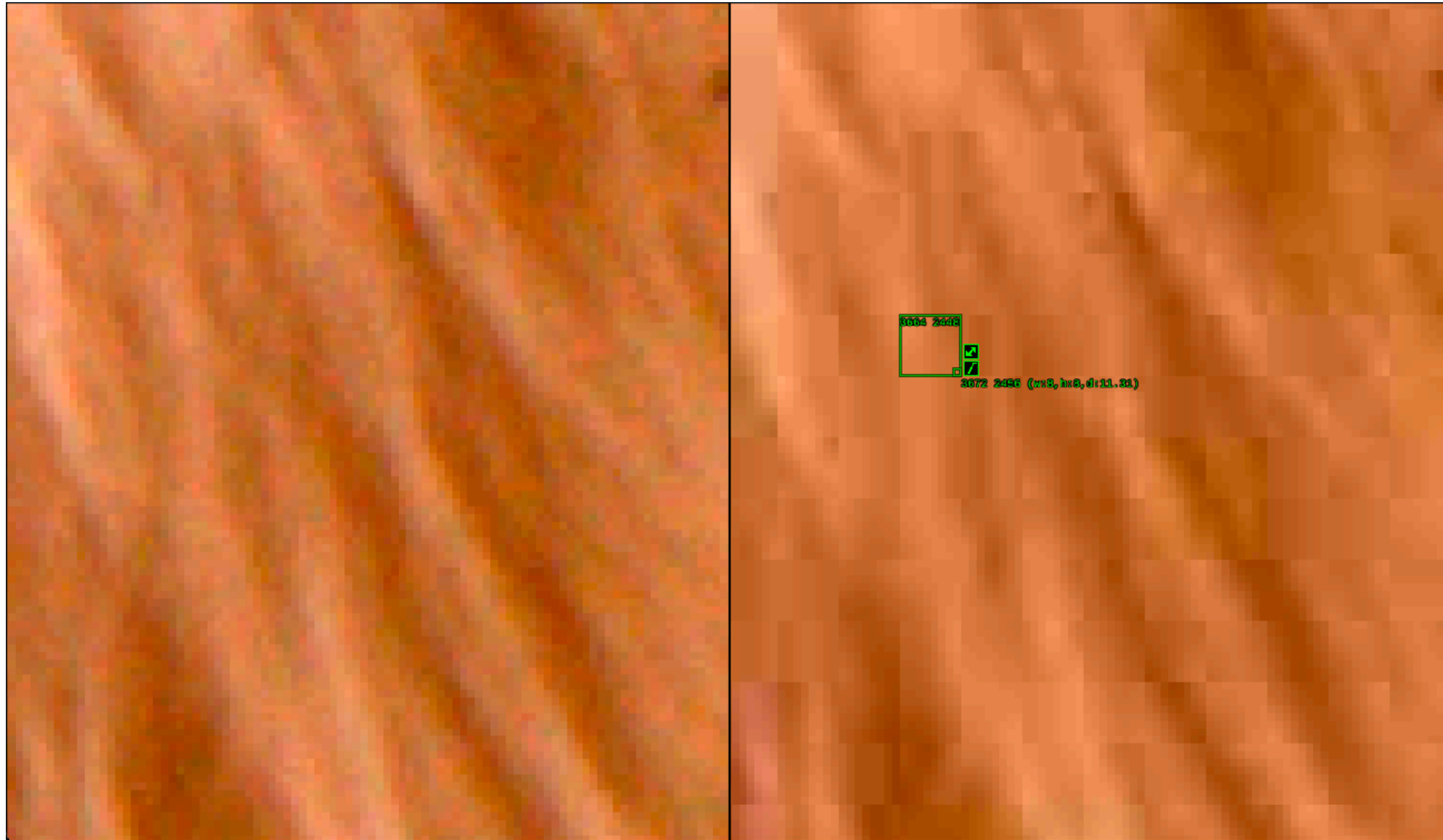
# JPEG grid artifact



Uncompressed image

Compressed image at QF = 50

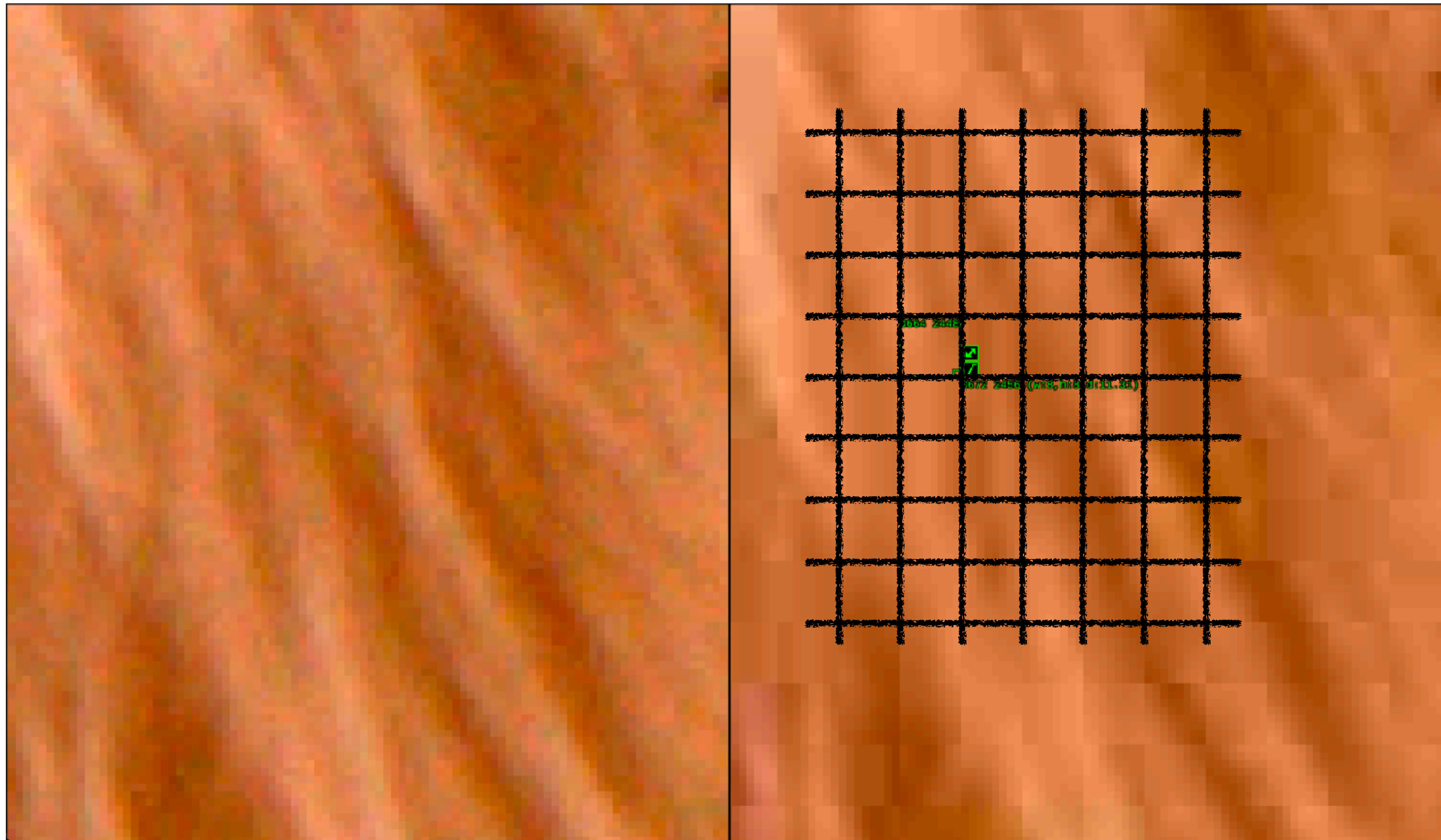
# JPEG grid artifact



Uncompressed image

Compressed image at QF = 50

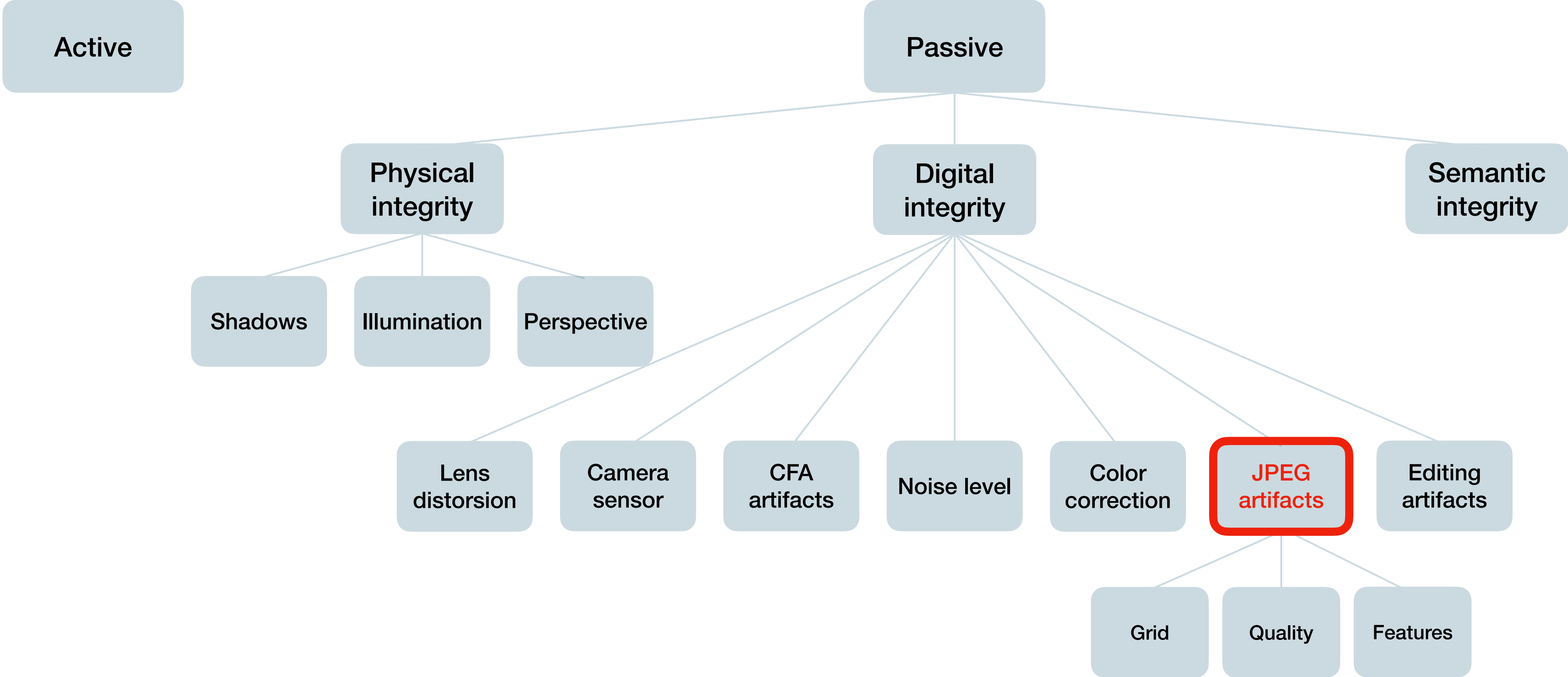
# JPEG grid artifact



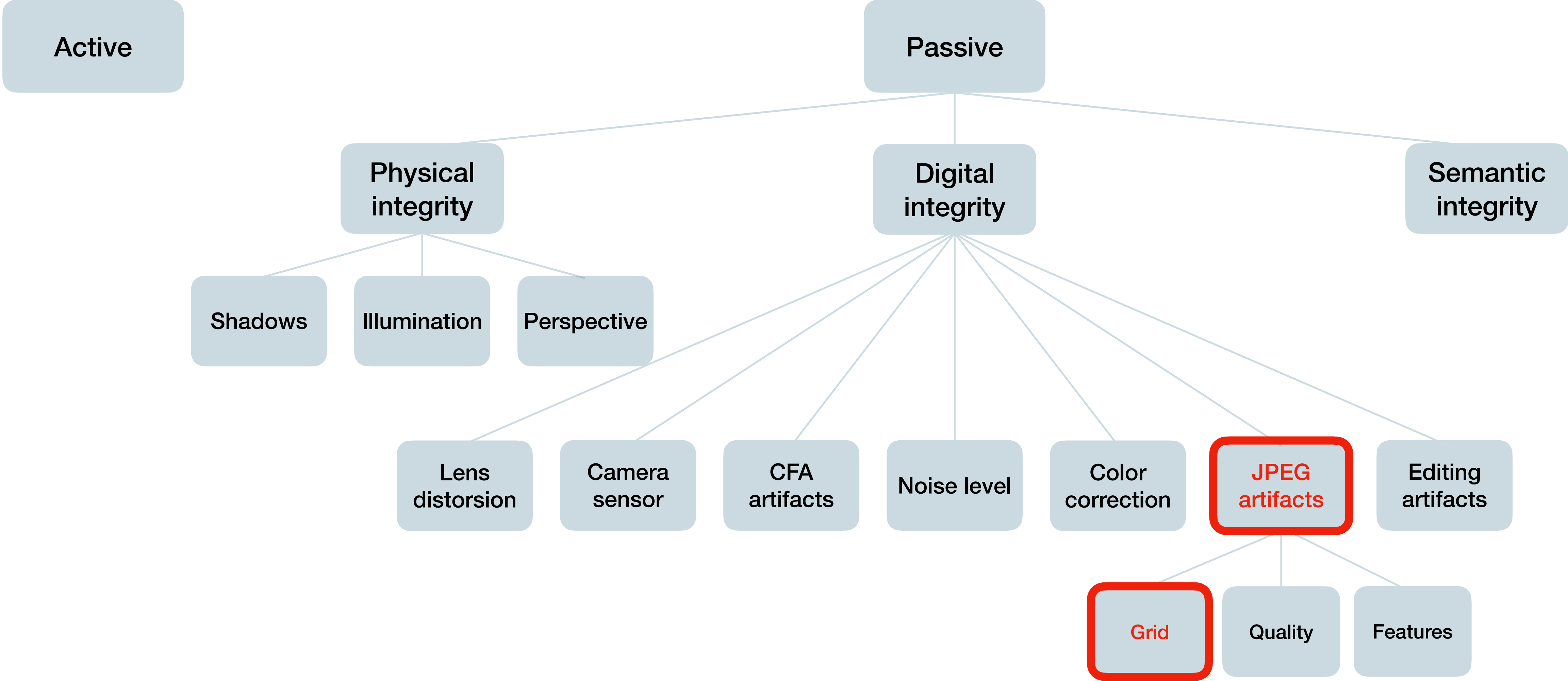
Uncompressed image

Compressed image at QF = 50

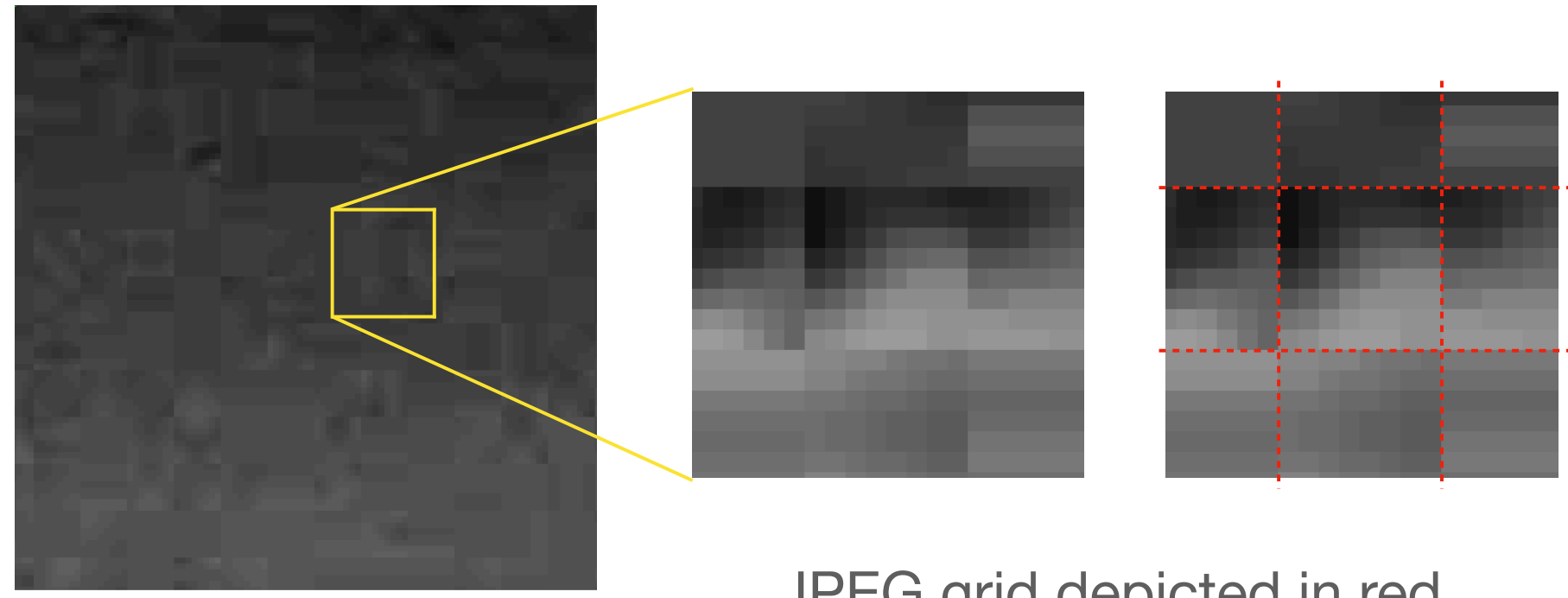
# Where do we stand?



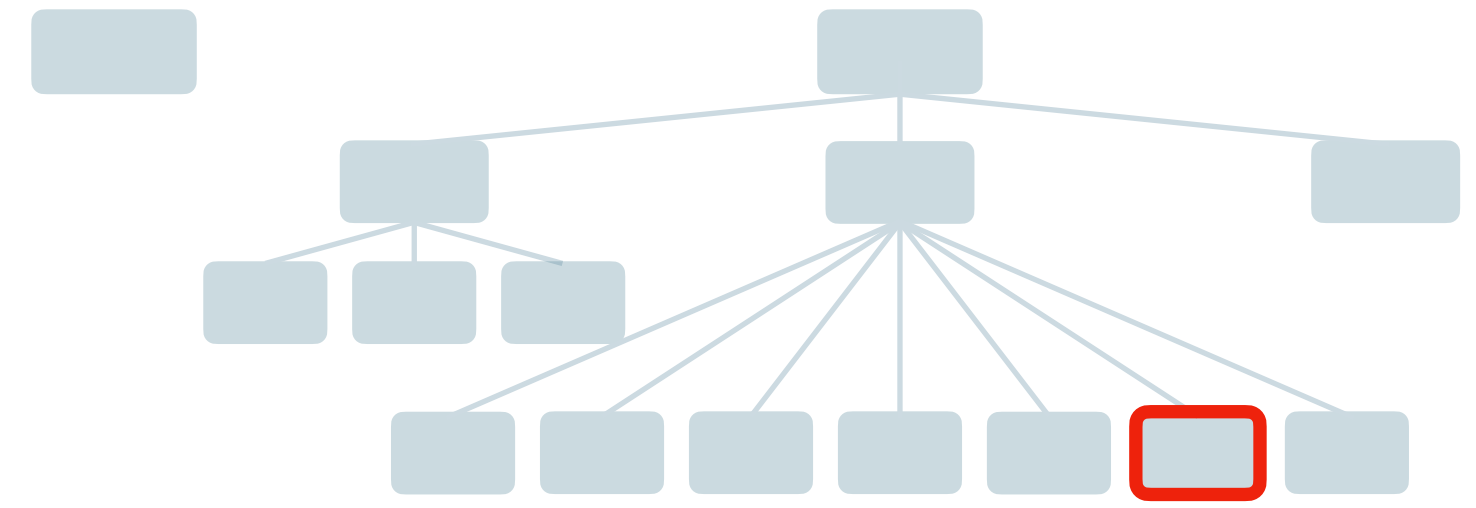
# Where do we stand?



# JPEG grid detection



JPEG grid depicted in red.



An overview on image forensics

## JPEG grid

### Block artifacts

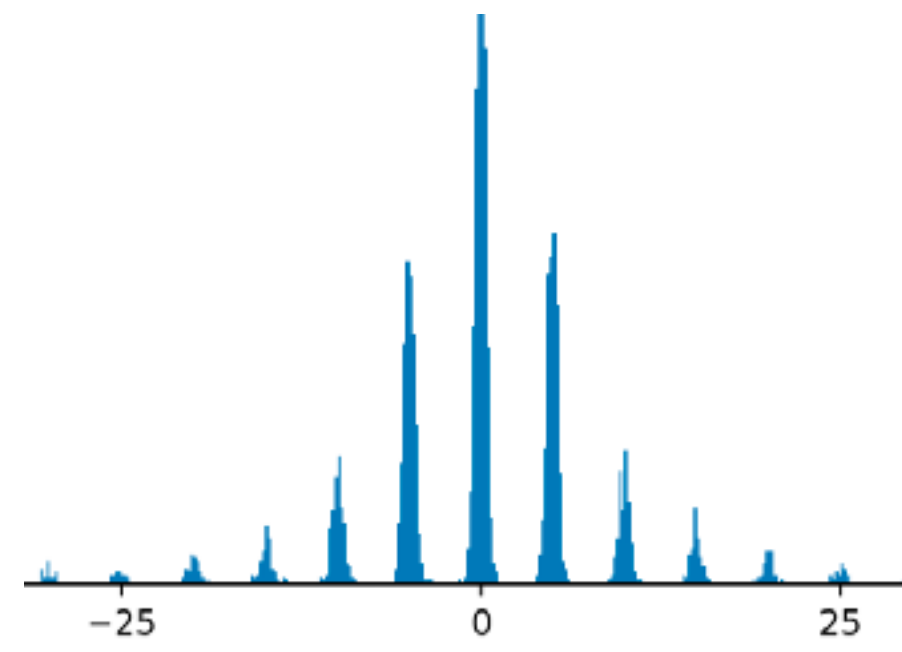
Chapter 2



Cross-difference image [Y.L. Chen et al. 2008].

### DCT coefficients

Chapter 3



Histogram of a DCT coefficient.

### File sizes

Chapter 4



### DCT zeros

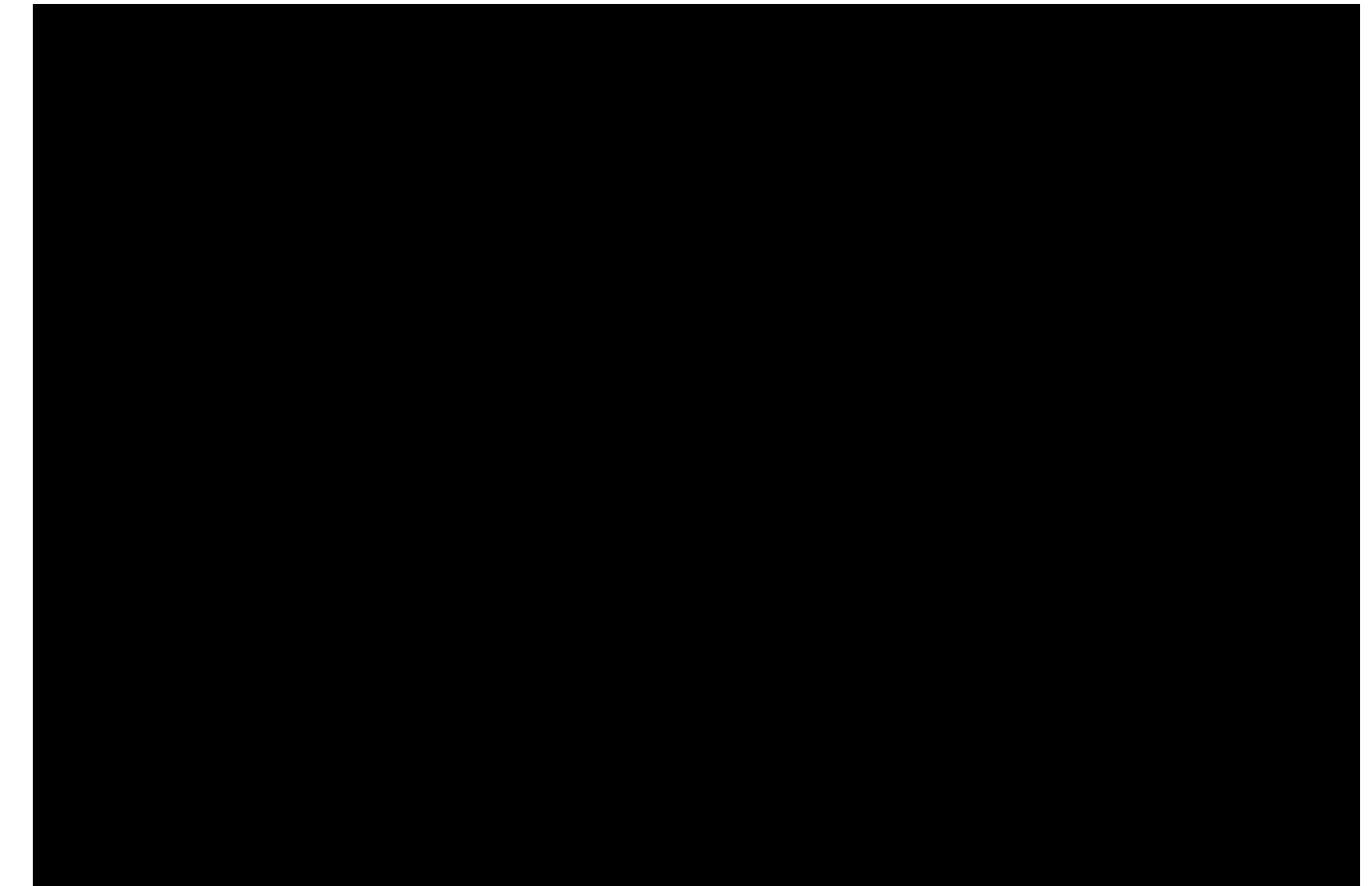
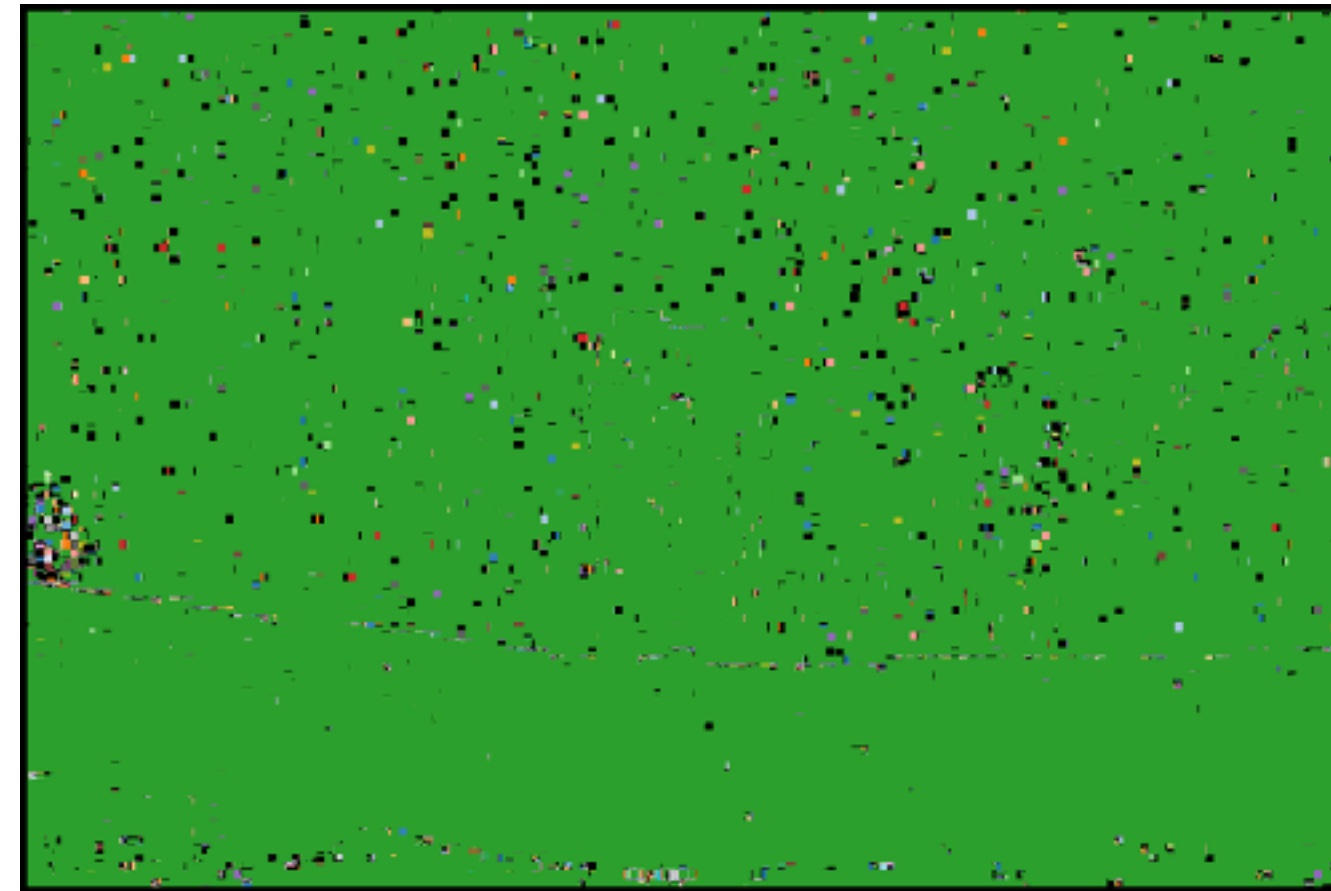
Chapter 5



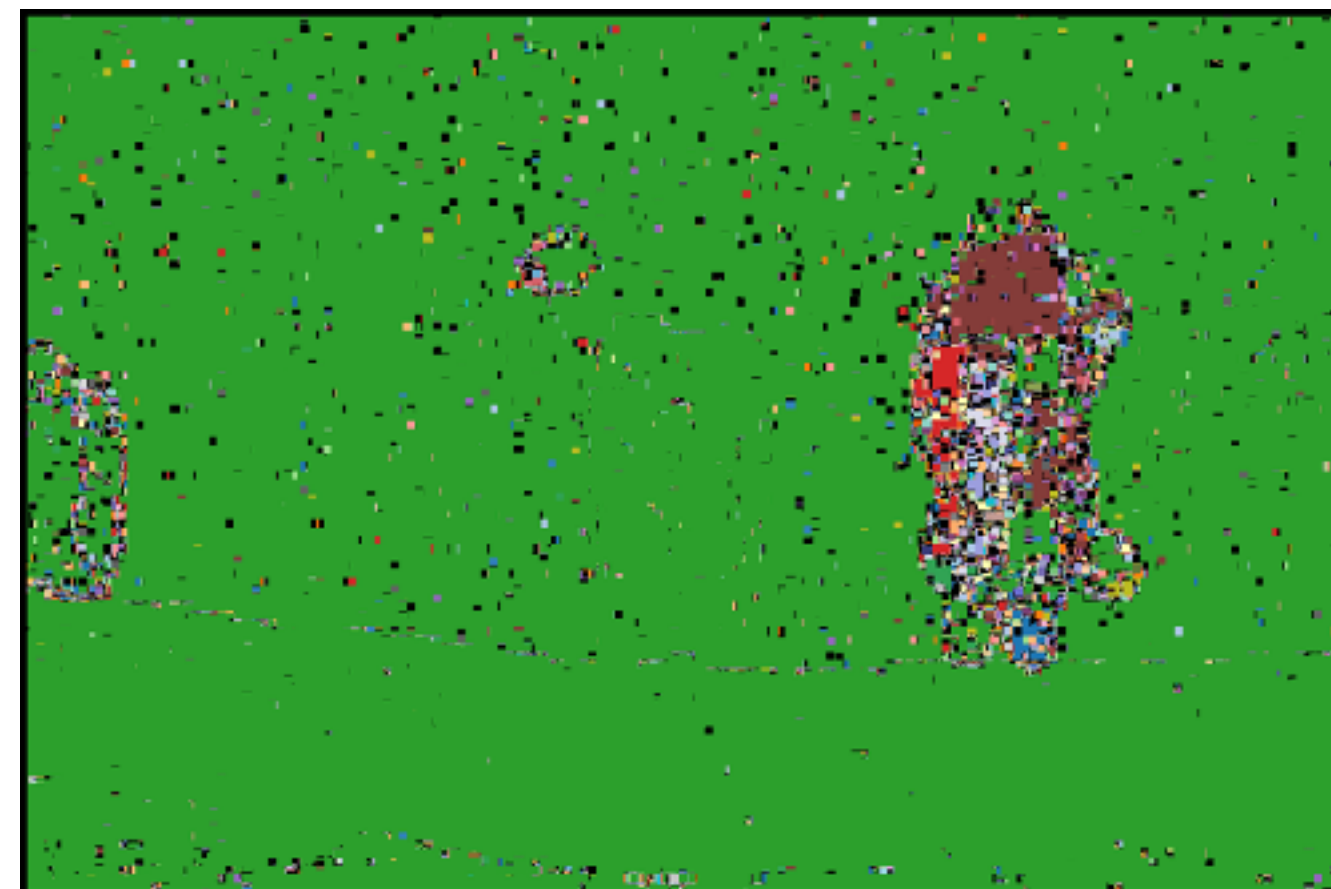
# ZERO

Automatic JPEG grid-based algorithm with controlled false alarms.

Original image



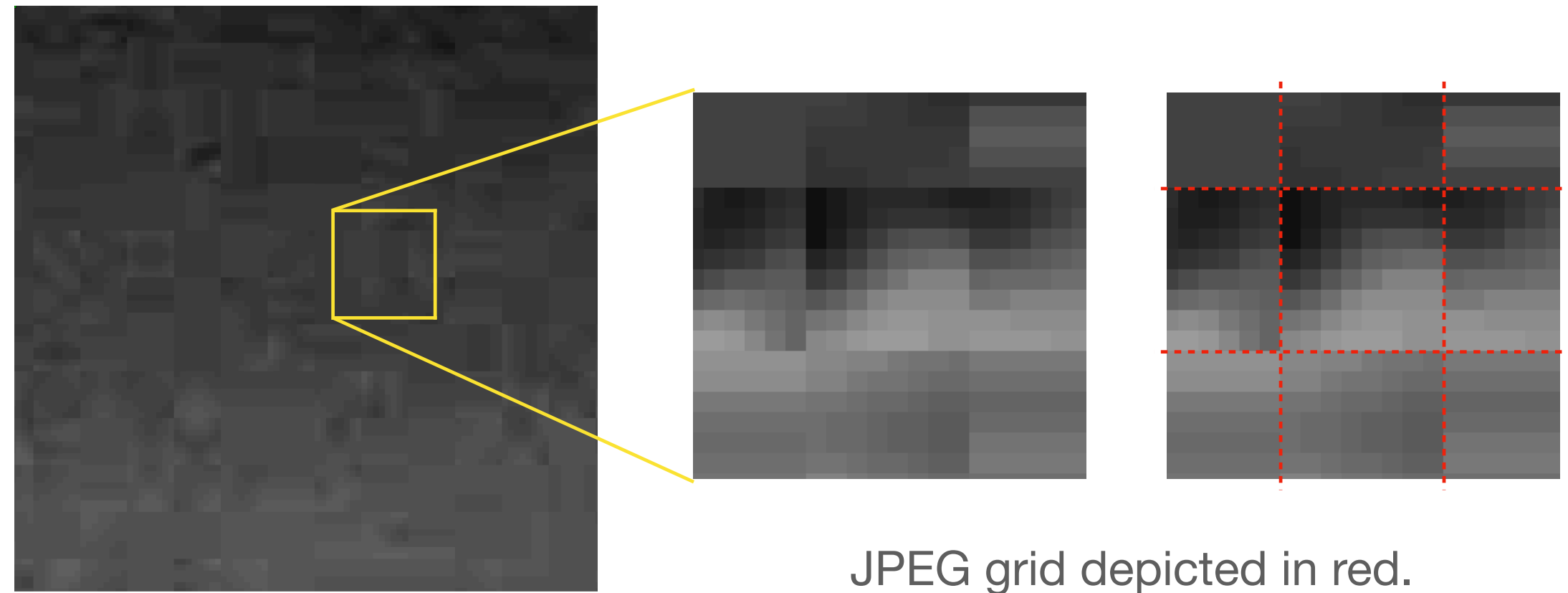
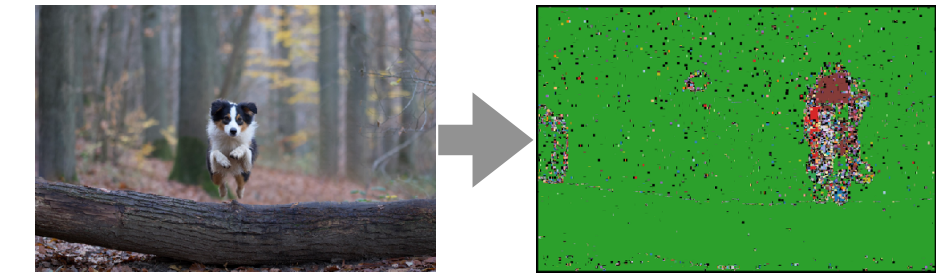
Forged image



Vote maps

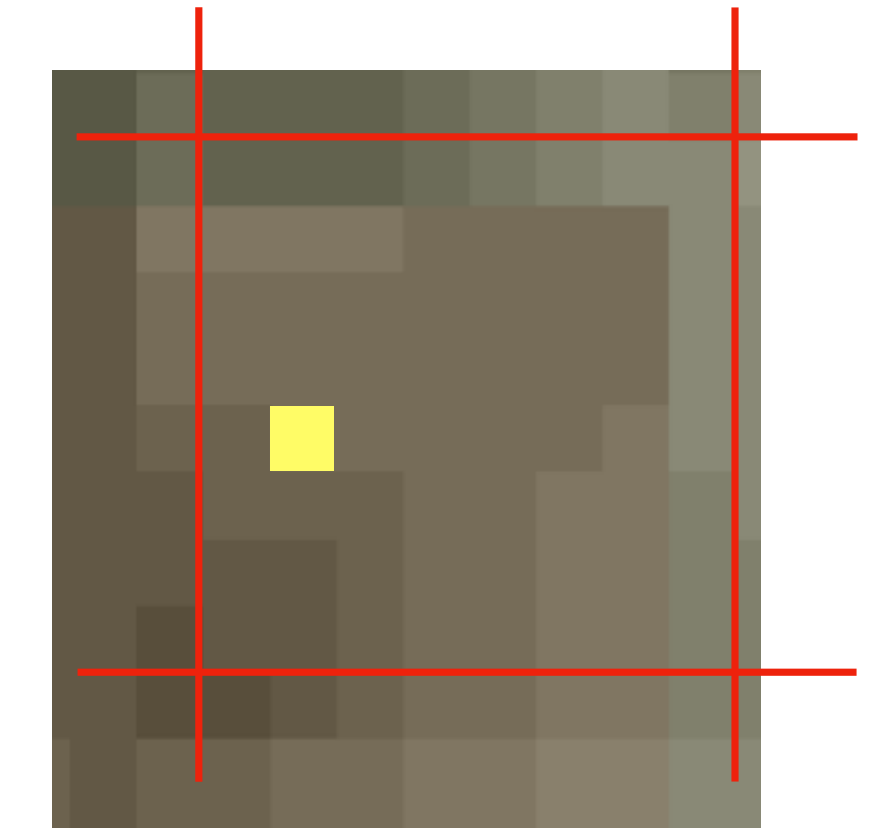
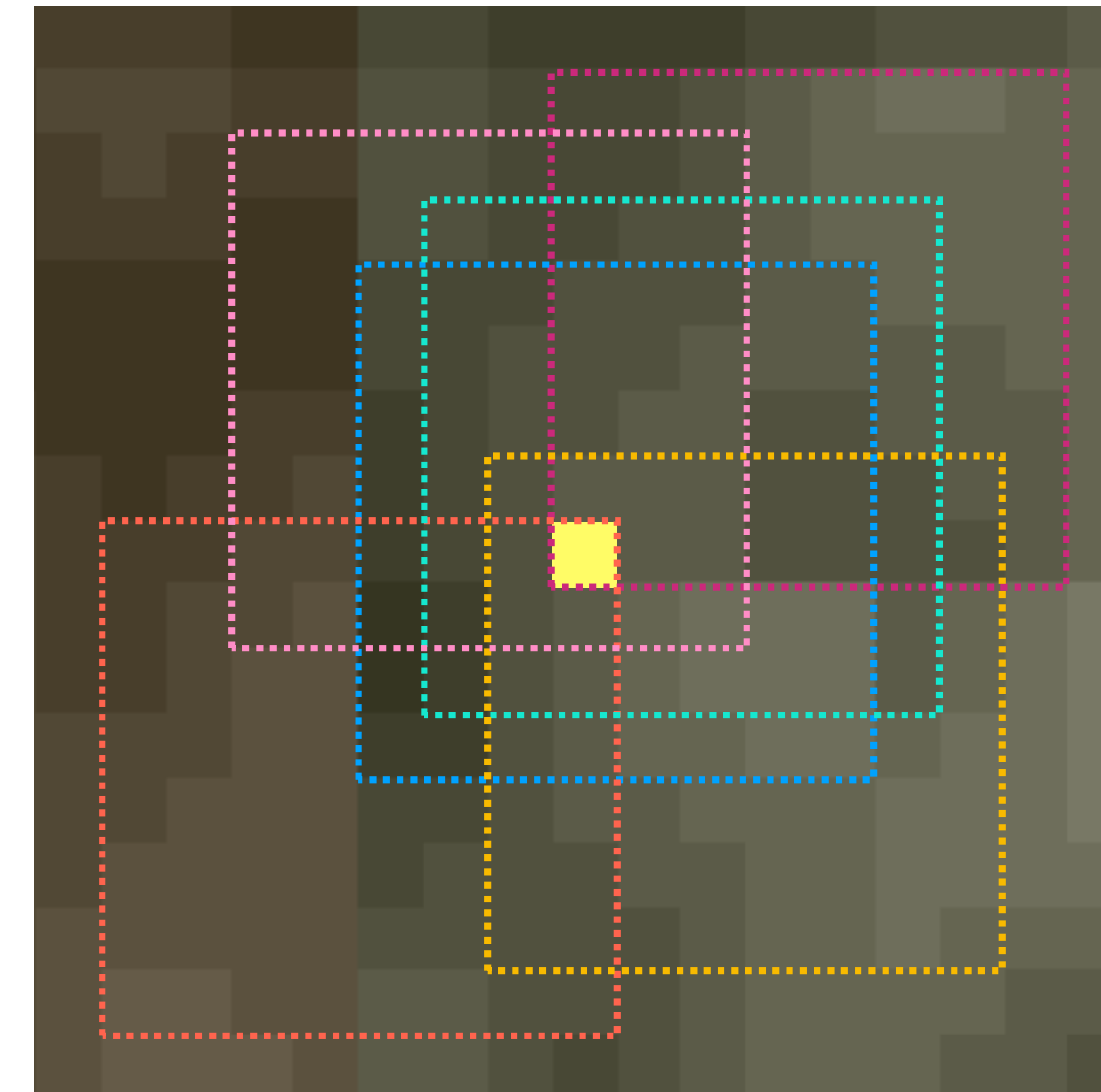
Detection results

# Voting process

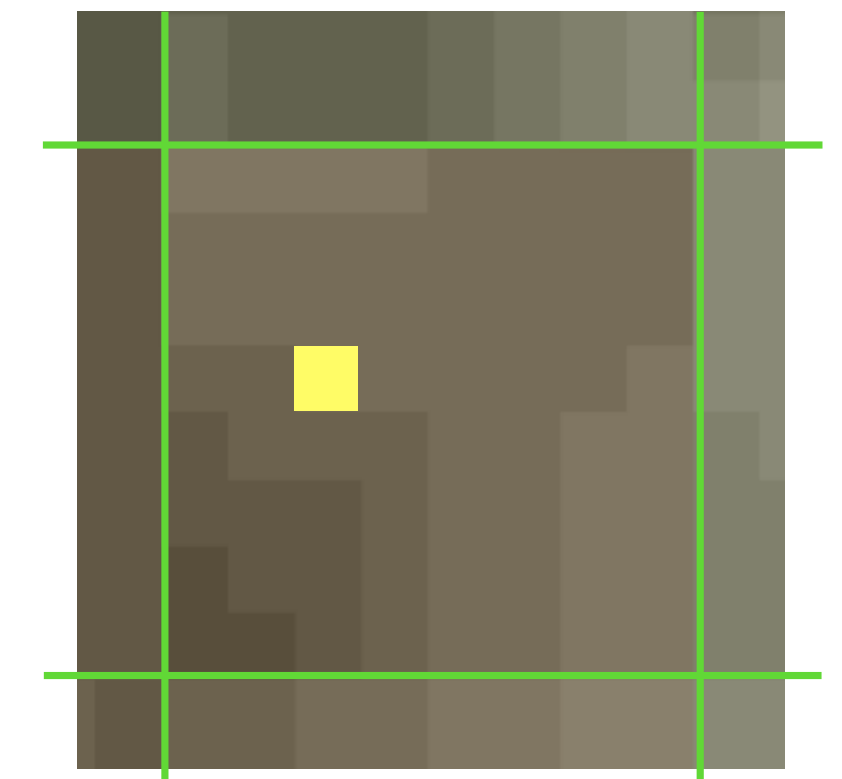


JPEG grid depicted in red.

A grid  $g$  is defined by the coordinates of its origin  $(g_x, g_y)$ .



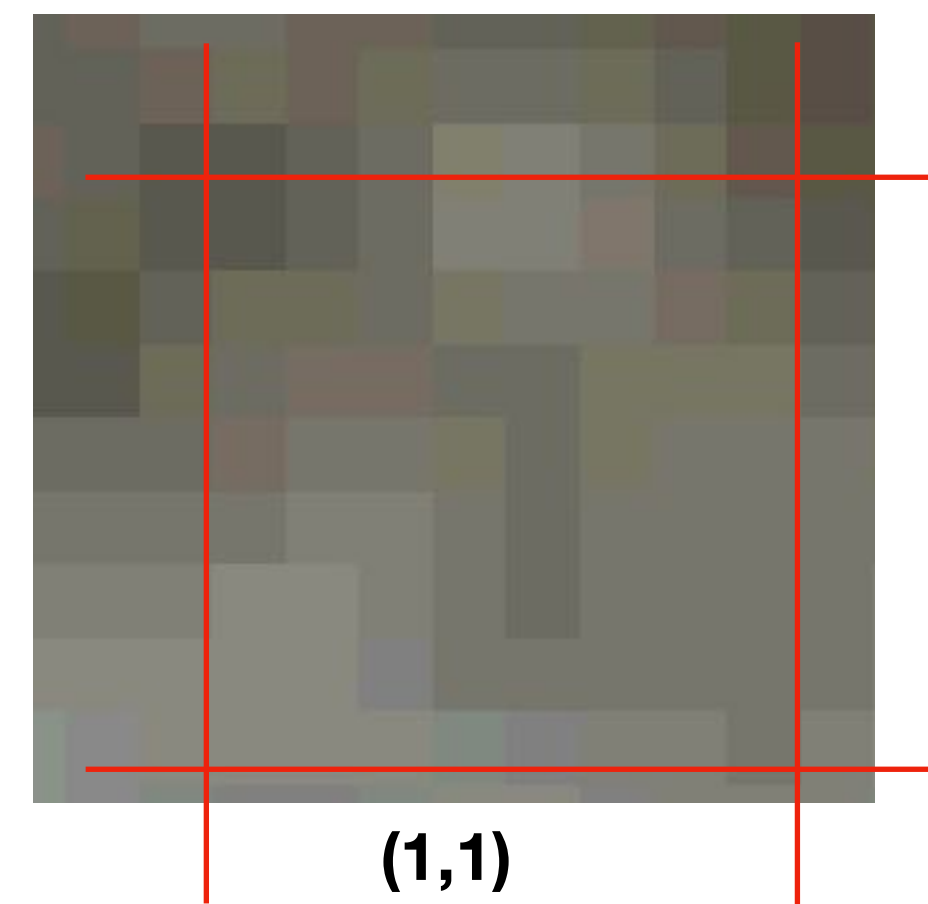
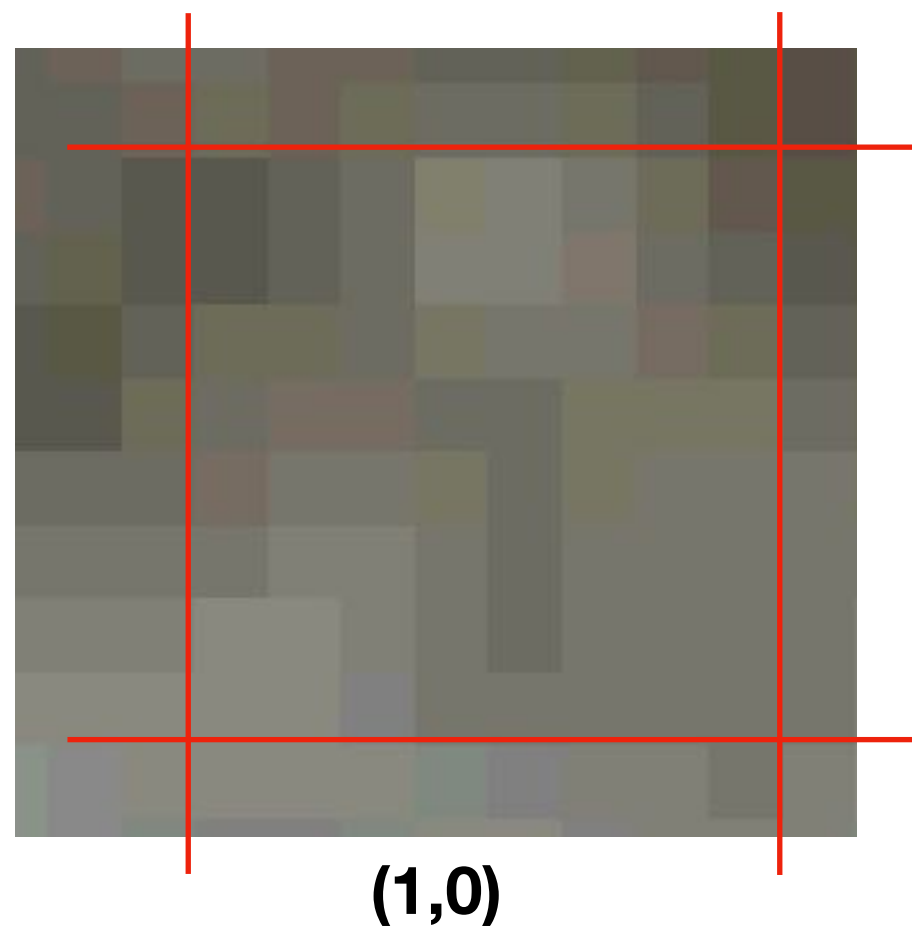
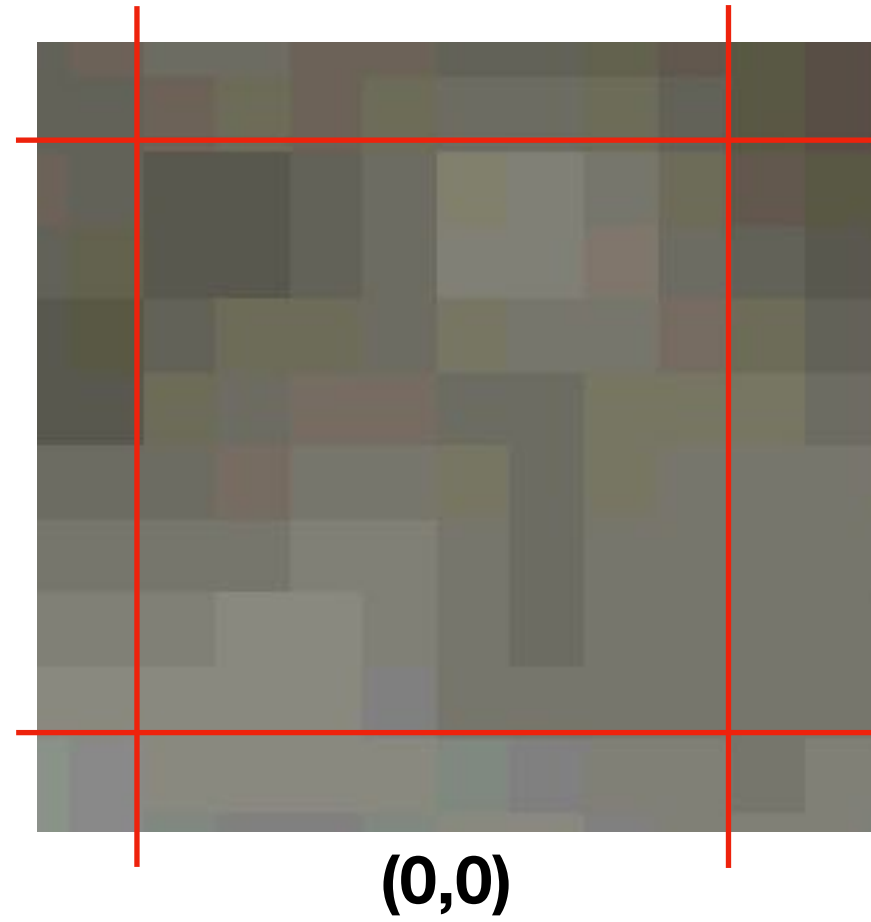
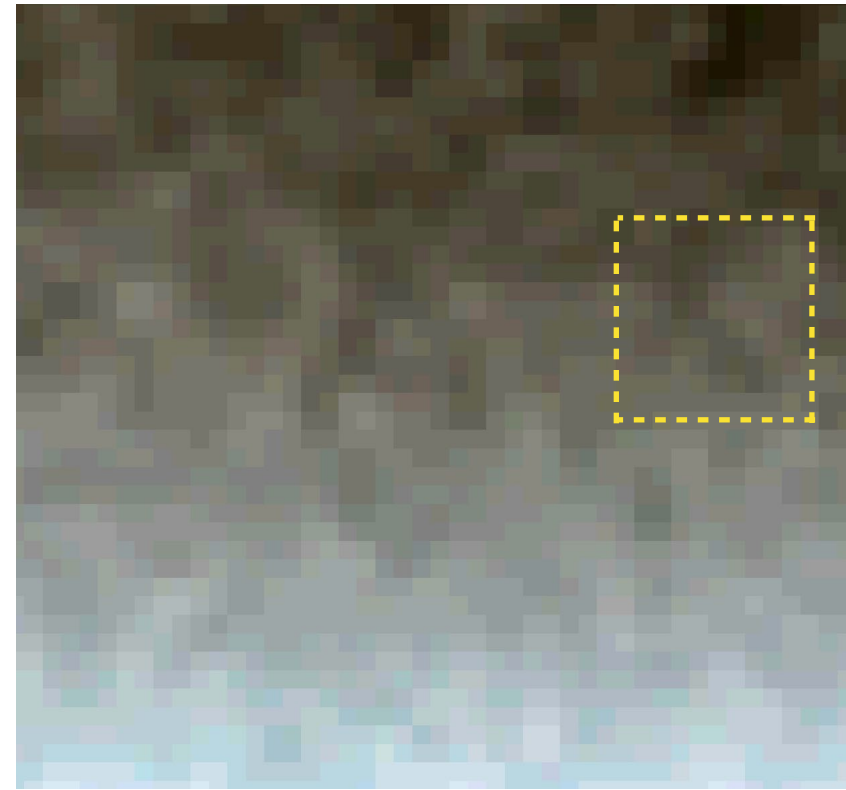
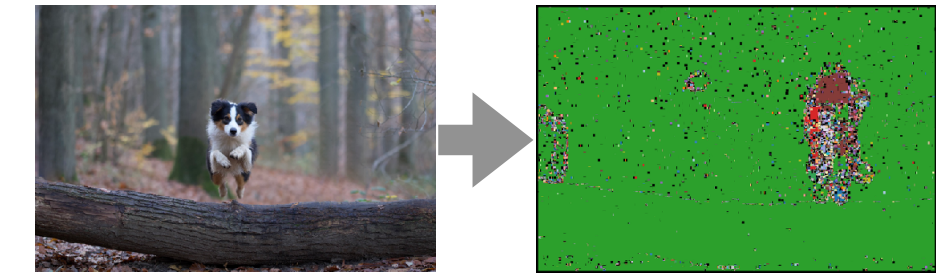
$(1,7)$



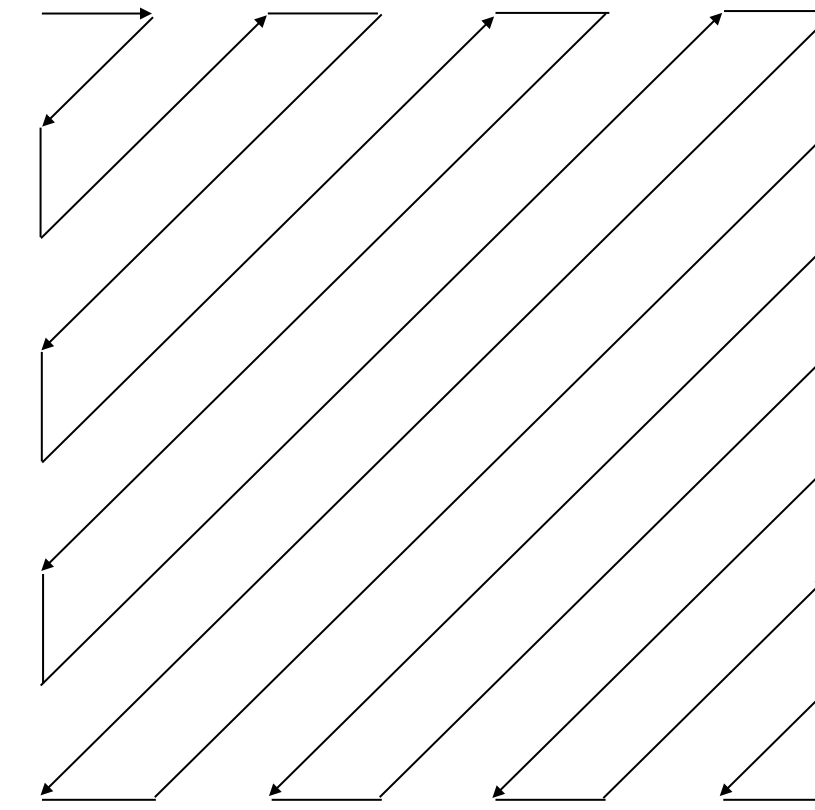
$(0,0)$

A pixel votes for the grid origin it belongs to.  
There are 64 different candidates.

# Voting process



Uncompressed image



641	-6	-15	1	-1	3	4	1
-34	-2	3	0	3	-1	-3	3
-15	1	4	2	-2	-3	2	-2
-7	3	4	-1	1	1	-1	2
-2	0	-4	-1	-2	4	0	-2
6	4	2	-3	1	-2	1	-1
4	2	-1	-2	1	2	0	-1
3	1	-1	-4	0	2	0	2

Uncompressed

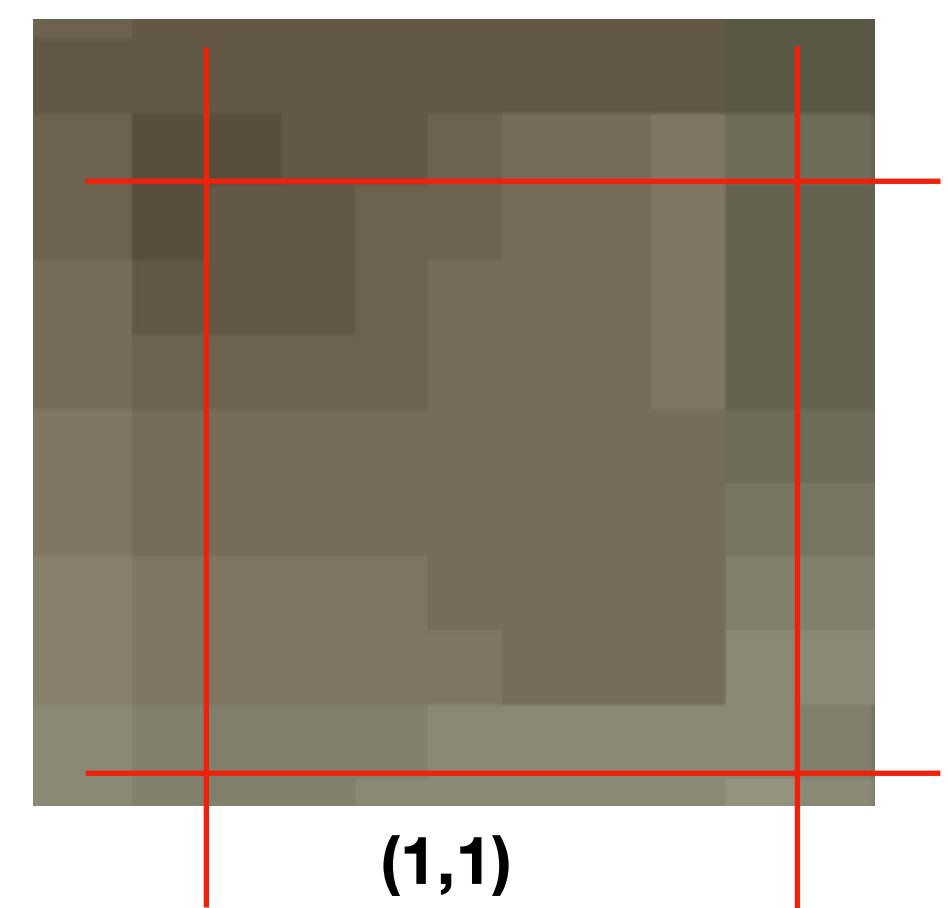
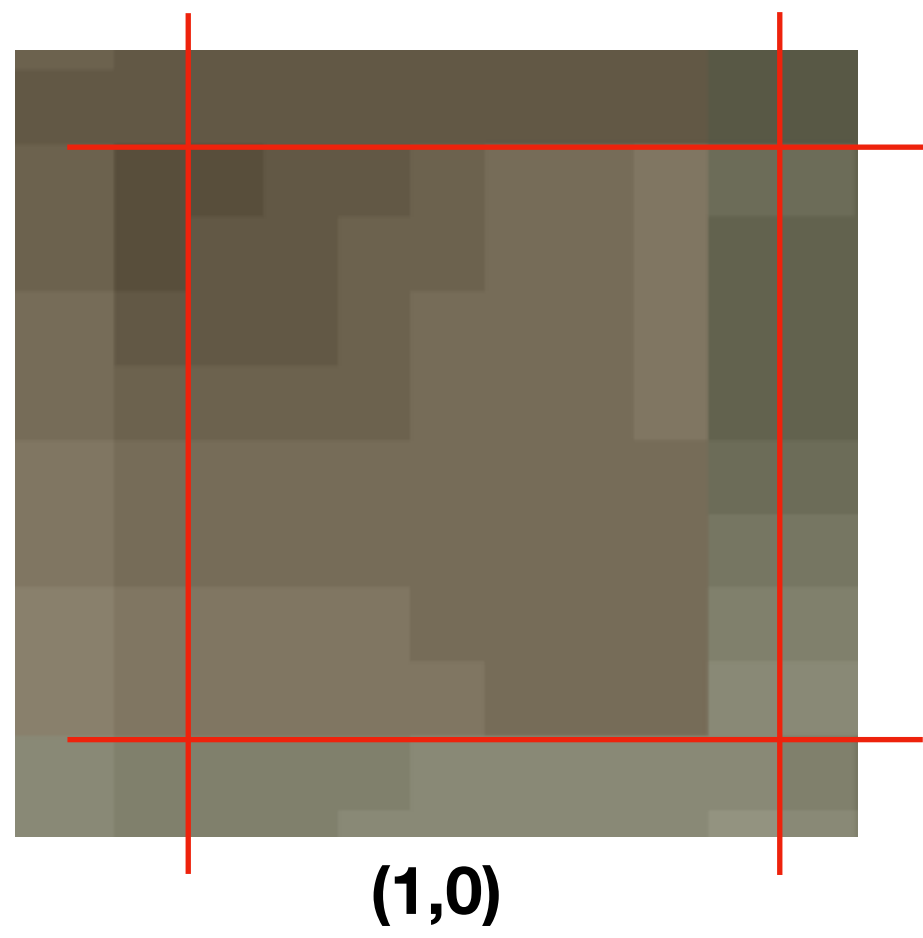
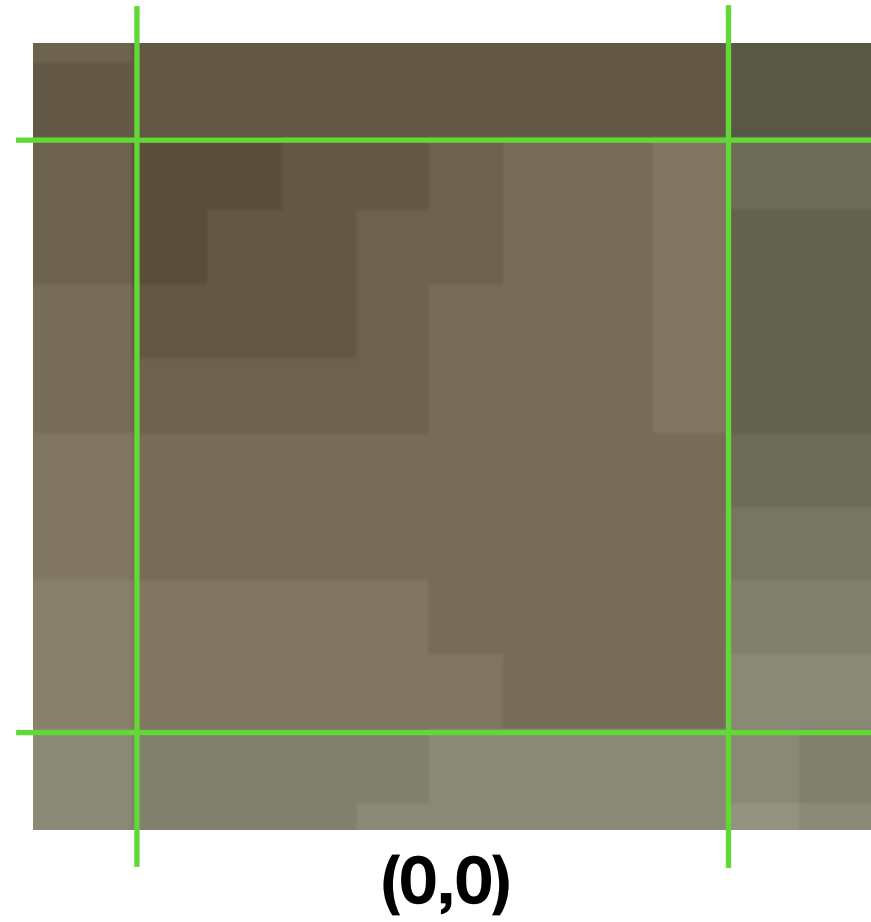
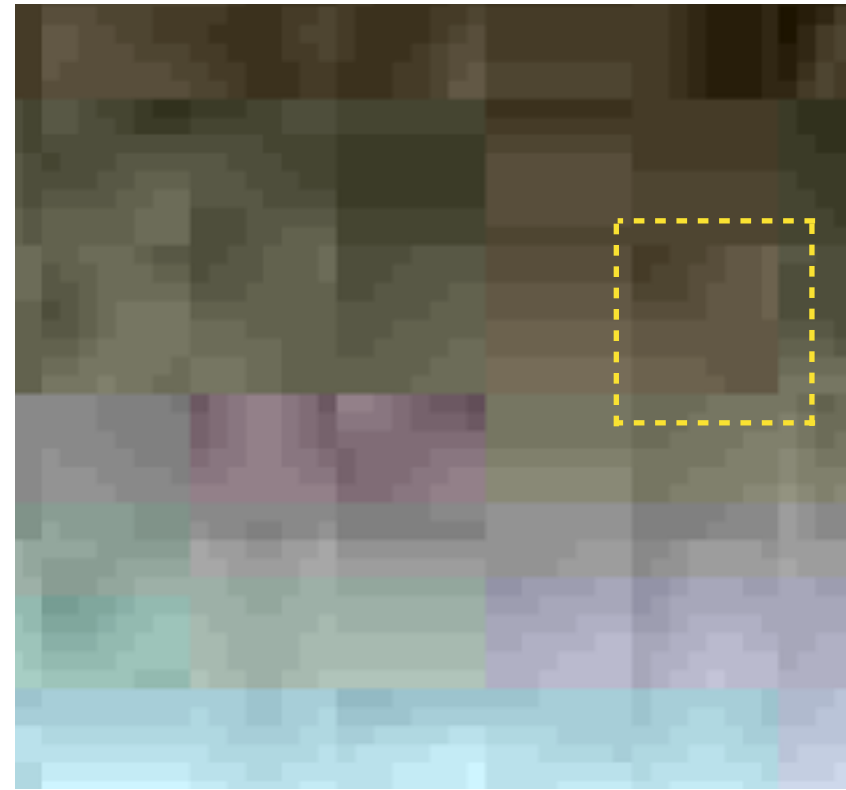
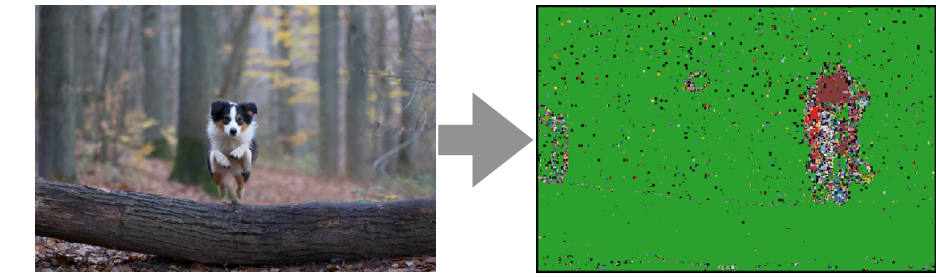
589	3	6	-3	0	2	-2	0
-16	6	0	4	-1	-1	-2	-2
32	-1	4	-4	-1	1	-2	0
-4	-3	4	1	-1	-2	-4	2
-12	-15	0	3	-2	-3	-1	-1
-21	-11	1	3	-2	-4	-1	1
-23	5	8	2	1	-2	-1	0
-6	7	2	0	2	-1	0	1

Uncompressed

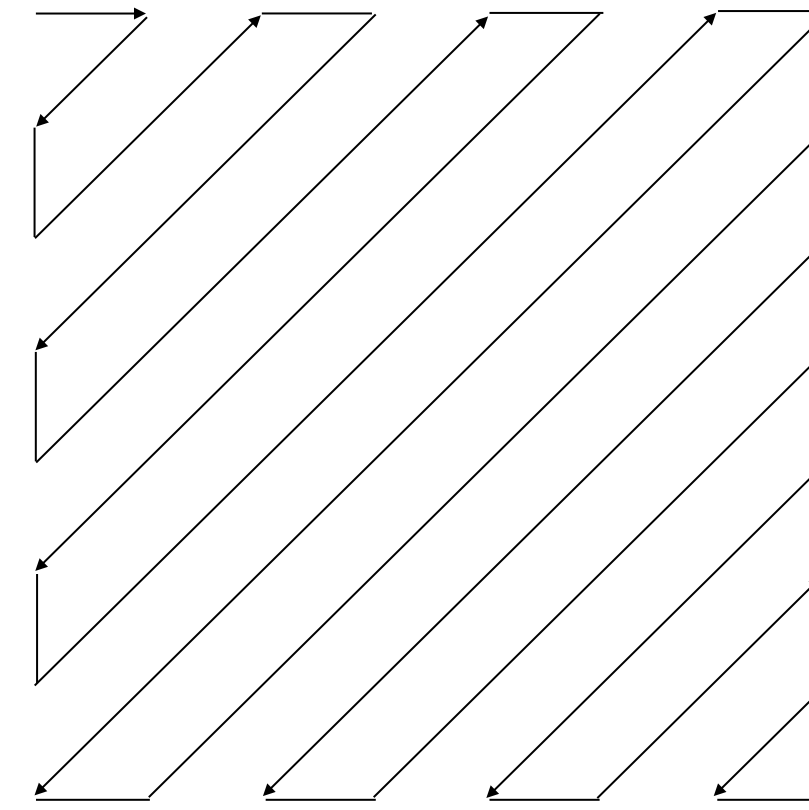
641	-6	-15	1	-1	3	4	0
-34	-2	3	0	3	-1	-3	0
-15	1	4	2	-2	-3	2	-2
-7	3	4	-1	1	1	-1	2
-2	0	-4	-1	-2	4	0	-2
6	4	2	-3	1	-2	1	-1
4	2	-1	-2	1	2	0	-1
3	1	-1	0	1	2	0	0

Uncompressed

# Voting process



Compressed image



586	-3	6	-1	0	1	-1	1
-15	5	-1	8	0	0	0	1
30	1	6	0	0	0	0	0
-7	-7	0	0	0	-1	0	0
-14	-18	1	0	0	0	0	0
-20	-14	0	0	0	0	0	0
-20	0	0	0	0	0	0	0
0	0	0	-1	0	0	0	0

Compressed correct position

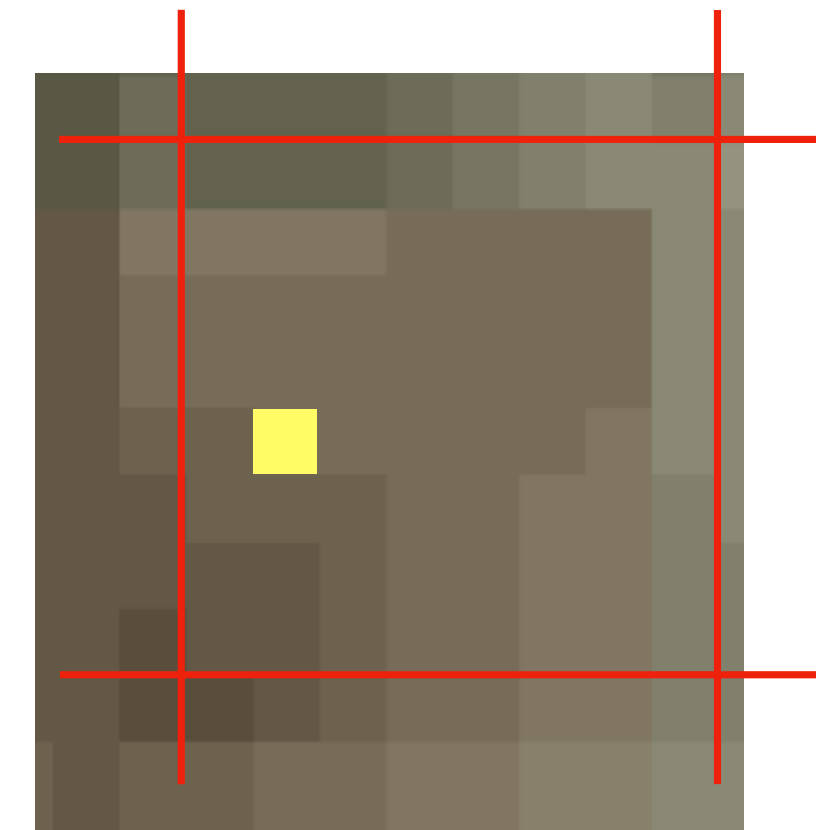
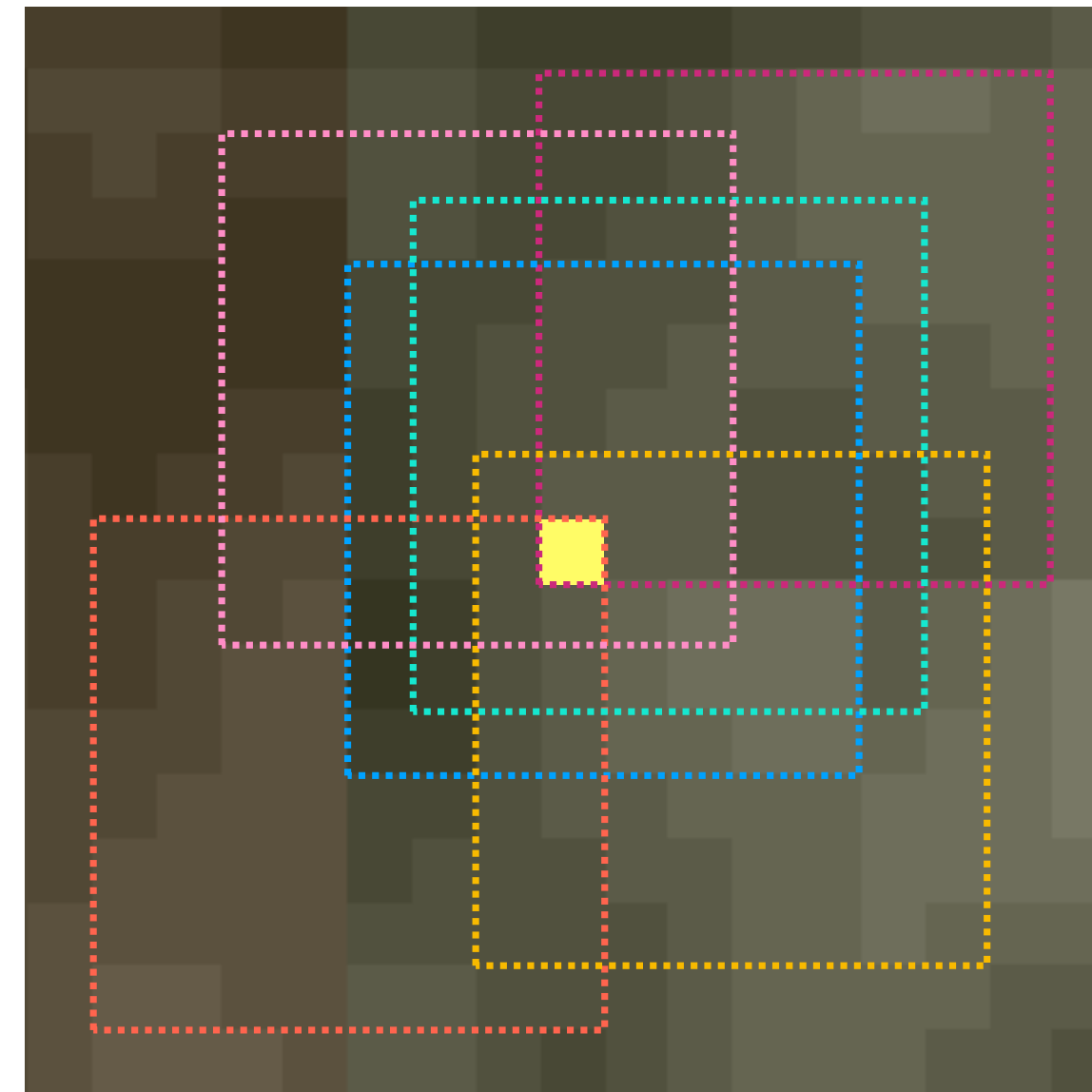
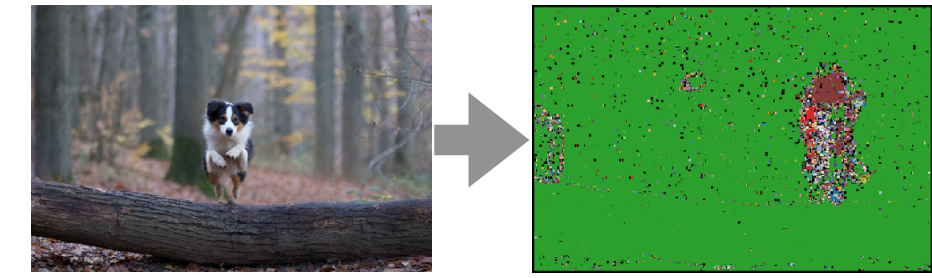
589	3	6	-3	0	2	-2	0
-16	6	0	4	-1	0	-2	-2
32	-1	4	-4	-1	1	-2	1
-4	-9	4	1	-1	-2	-4	0
-12	-15	4	0	-2	-3	-1	-1
-21	-11	7	3	-2	-4	-1	1
-23	5	8	2	1	-2	-1	0
-6	7	2	2	0	-1	0	1

Compressed wrong position

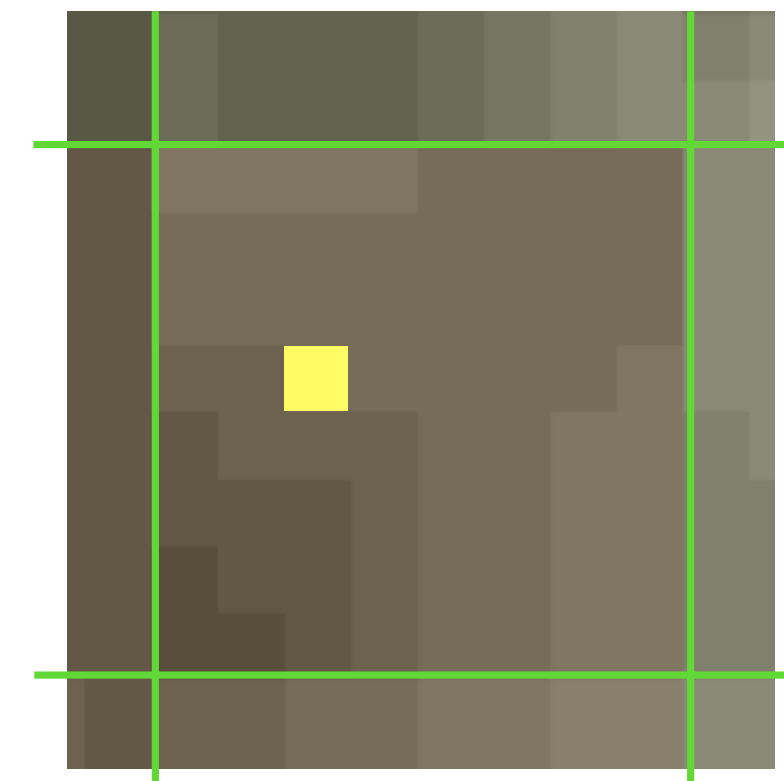
641	-6	-15	1	-1	3	4	0
-34	-2	3	0	3	-1	-3	1
-15	1	4	2	-2	-3	2	-2
-7	3	4	-1	1	1	-1	2
-2	0	-4	-1	-2	4	0	-2
6	4	2	-3	1	-2	1	-1
4	0	-1	-2	0	2	0	-1
3	1	-1	1	0	2	1	0

Compressed wrong position

# Voting process



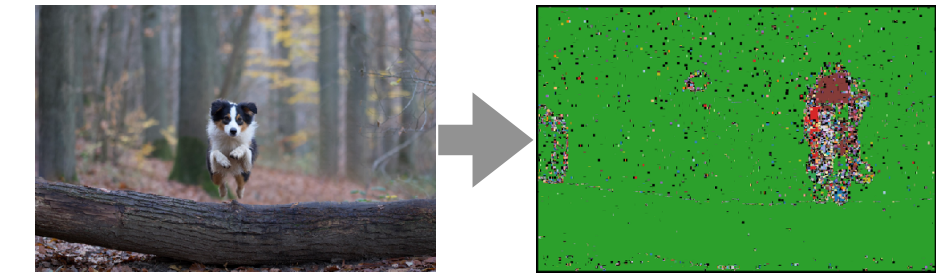
(1,7)



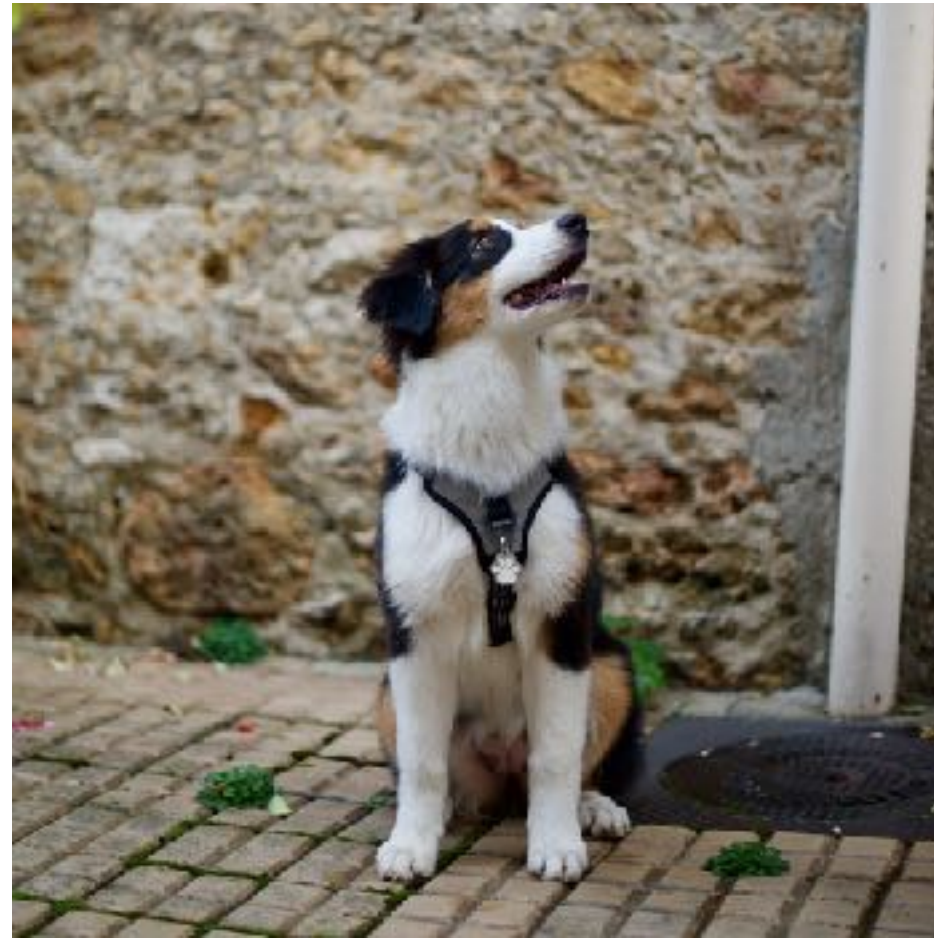
(0,0)

A pixel votes for the grid origin of the DCT block having the most zeros.  
There are 64 different candidates (64 different colors).

# Voting process



Uncompressed



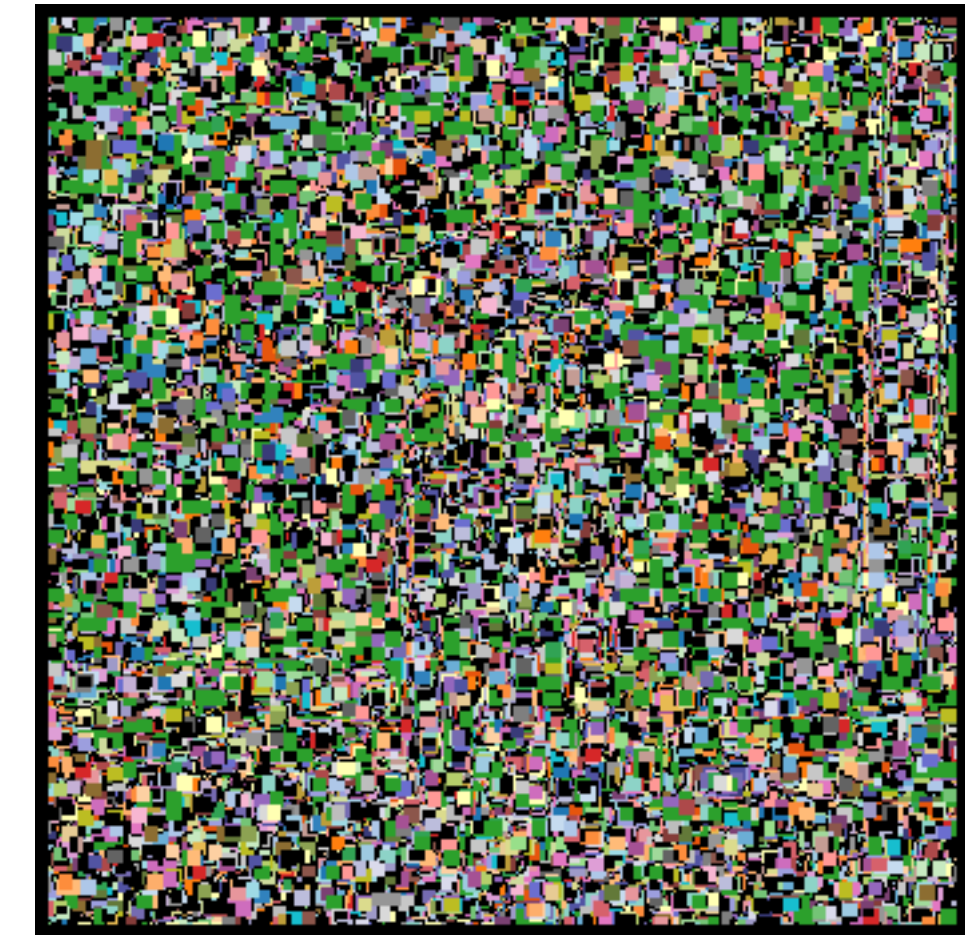
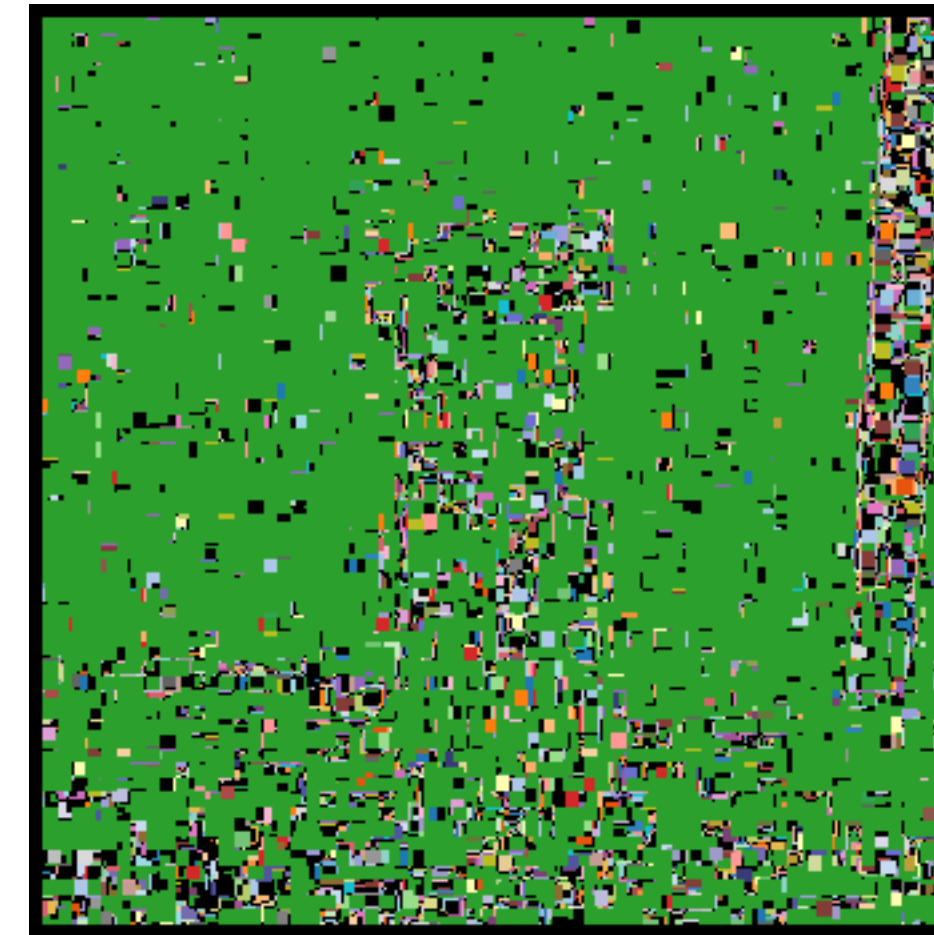
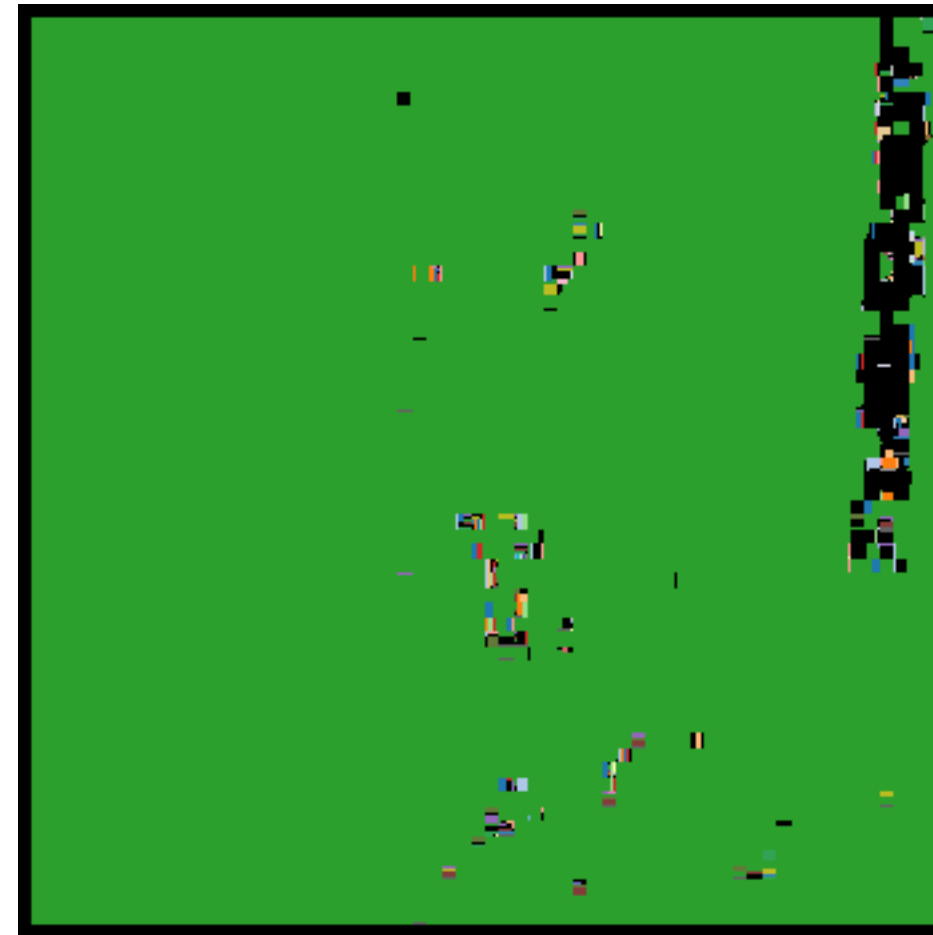
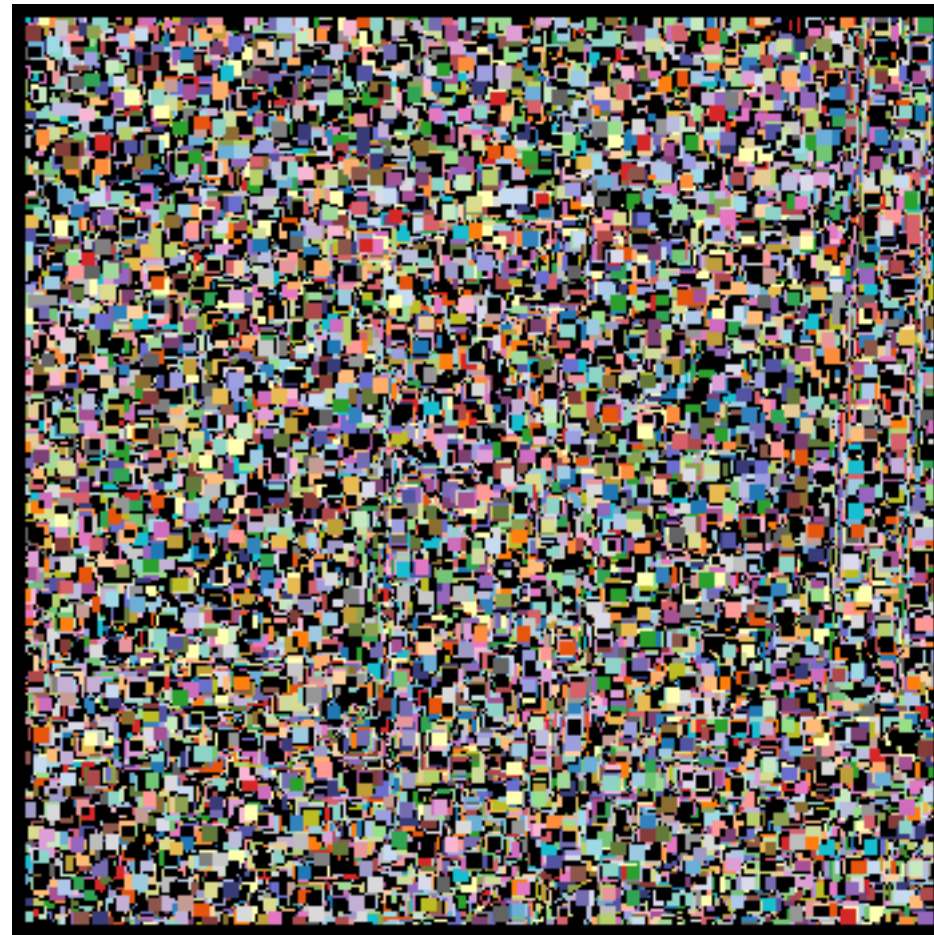
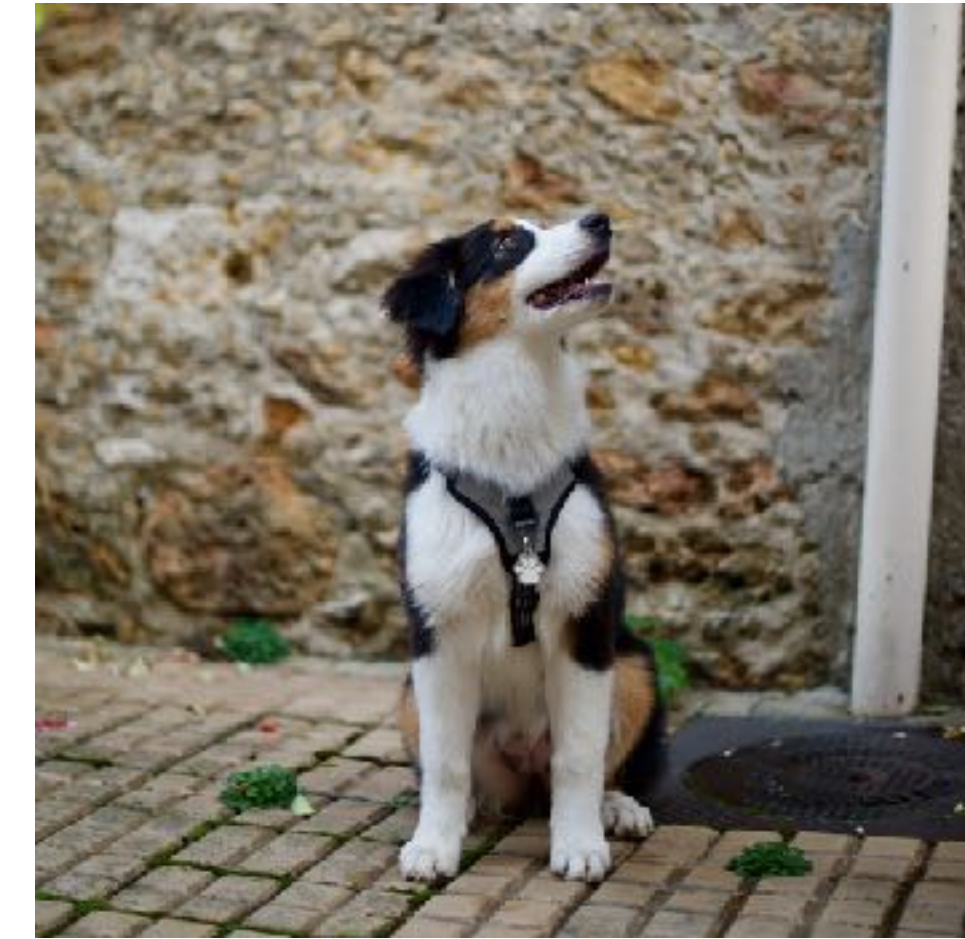
Compressed (QF = 80)



Compressed (QF = 98)

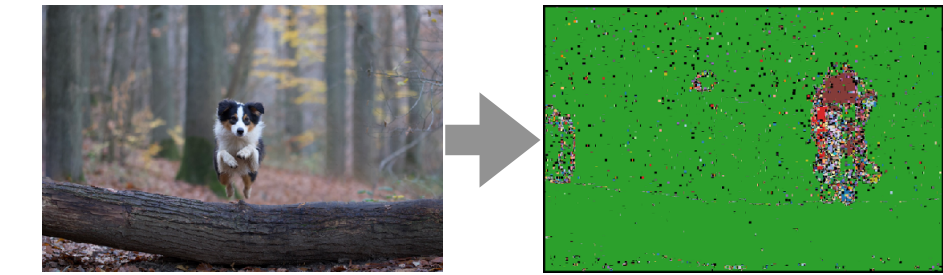


Compressed (QF = 99)

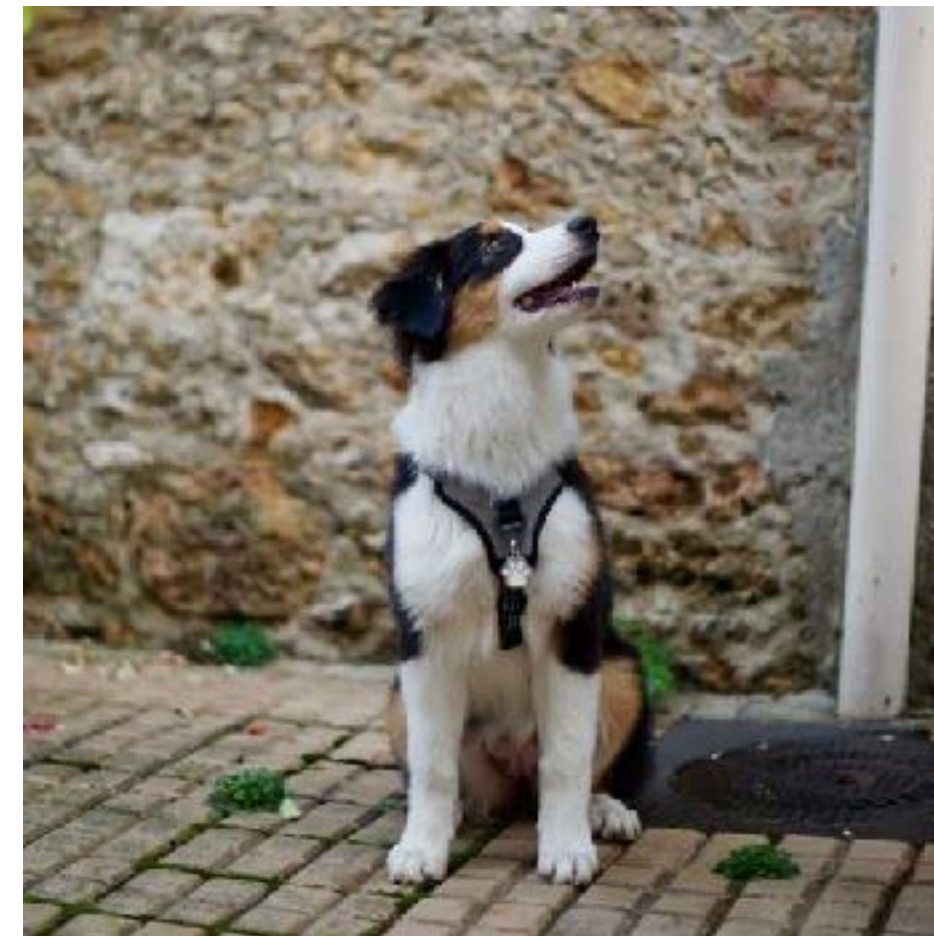


Each color represents a grid origin. Black corresponds to a non-valid vote, in case of a tie for example.

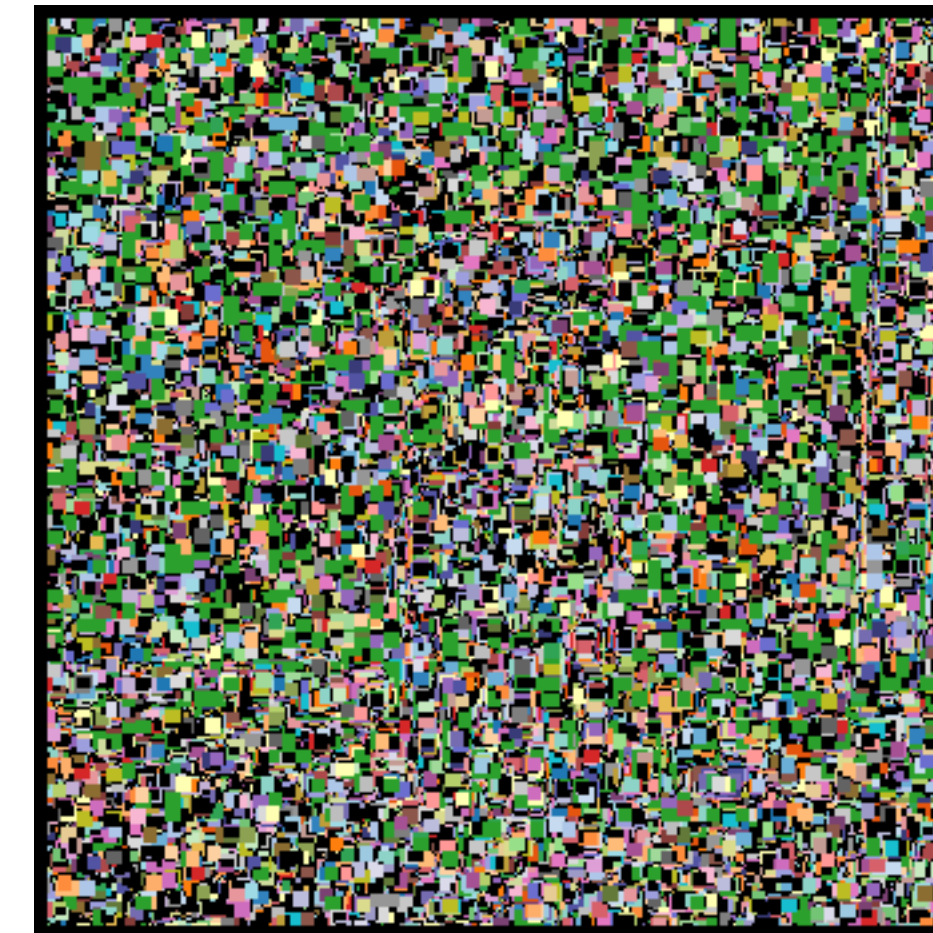
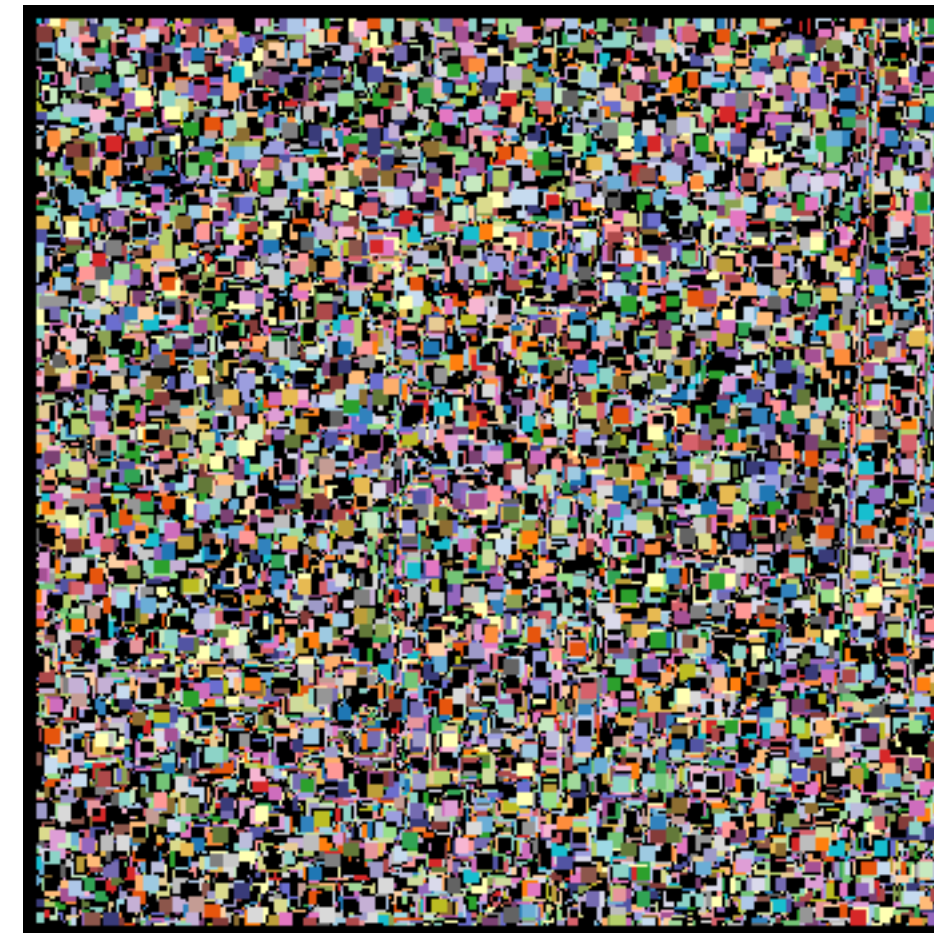
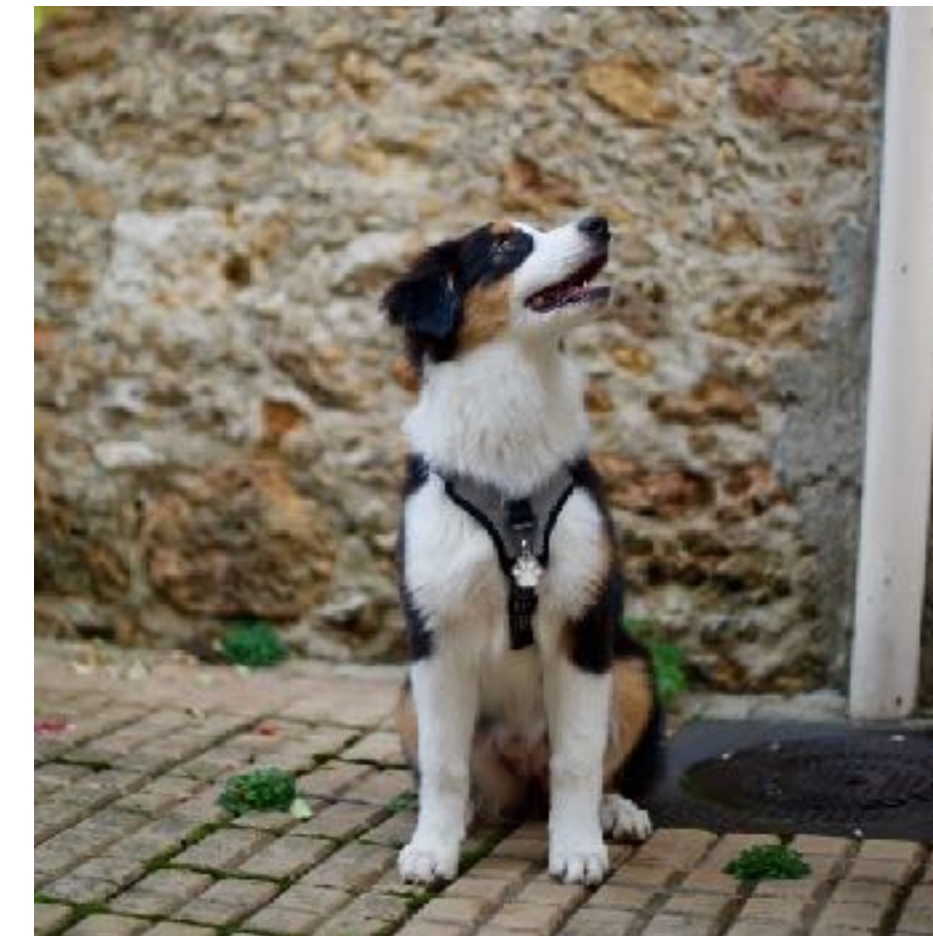
# How to decide?



Uncompressed

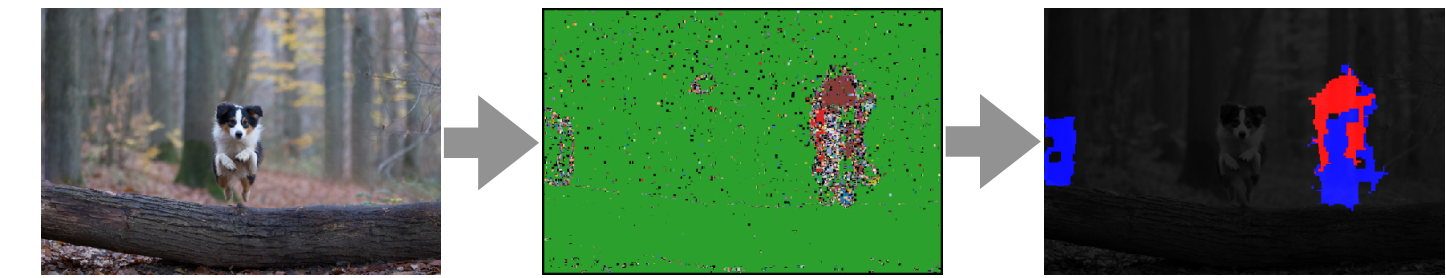


Compressed (QF = 99)



Each color represents a grid origin. Black corresponds to a non-valid vote, in case of a tie for example.

# Validation step

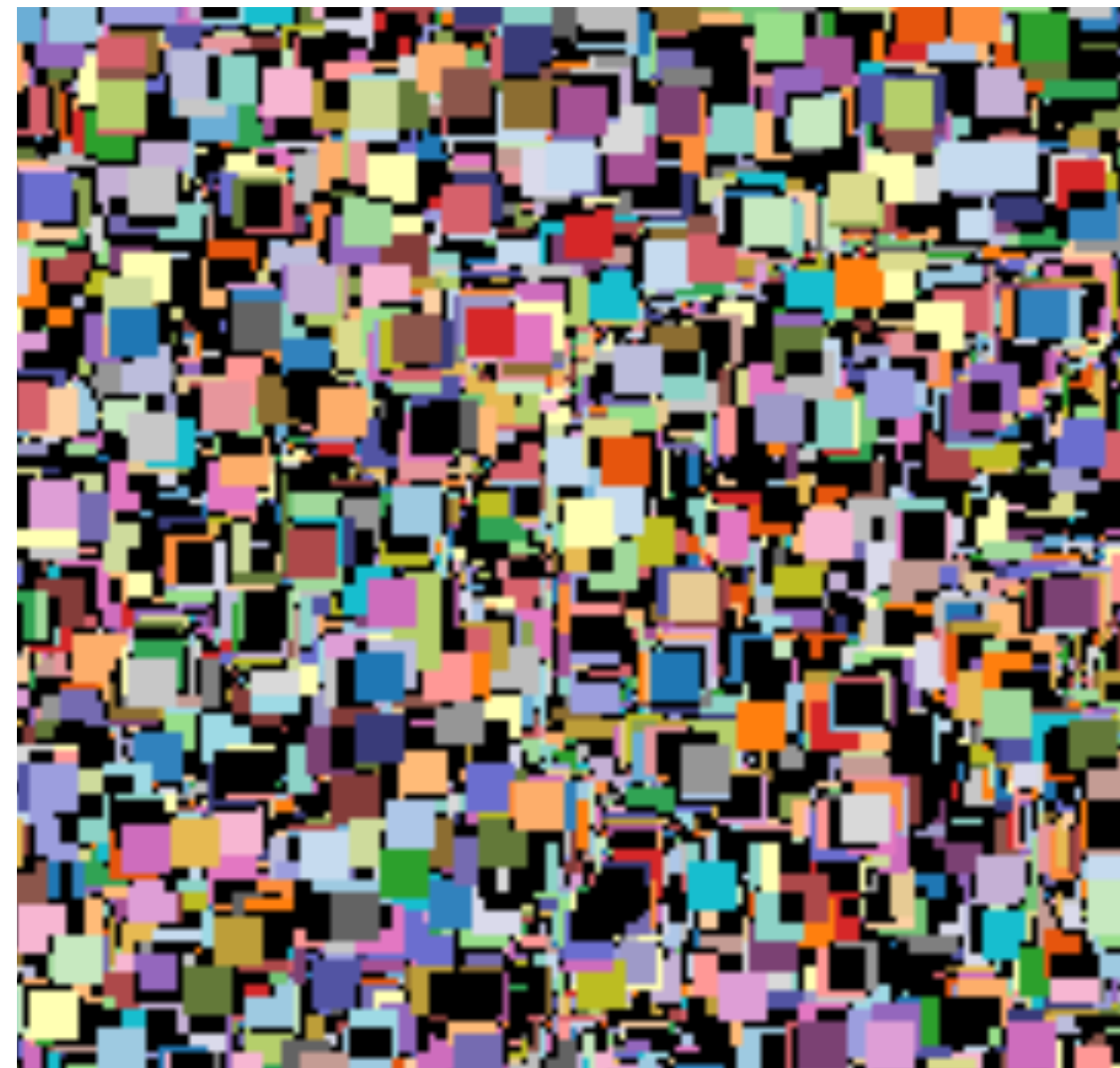


Given a window of an image, is a vote for the candidate (0,0) significant?

Among the  $n = 625$  votes,  
 $k = 30$  voted for (0,0)

4.8% votes

$> 1.56\% = 1/64$

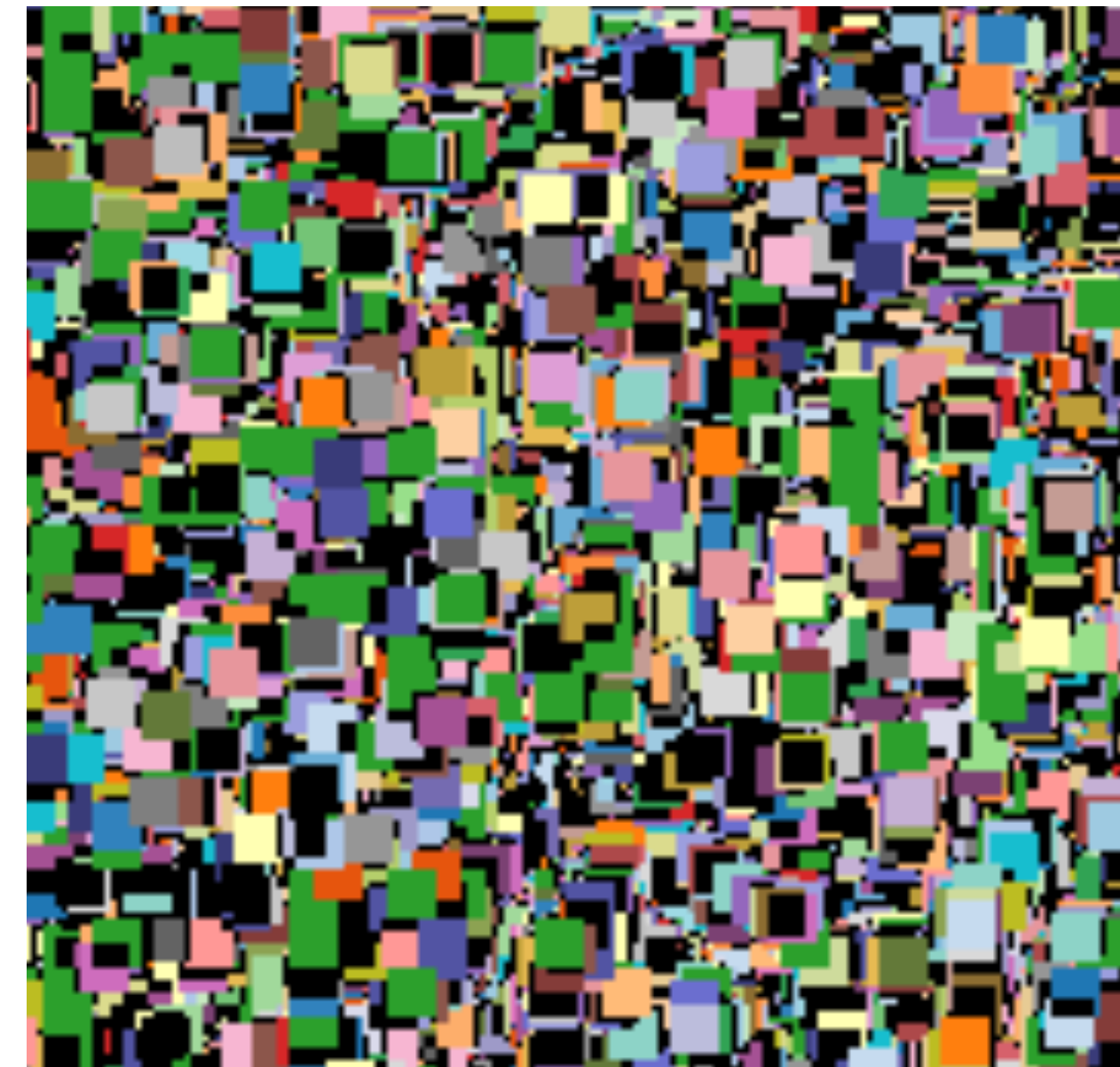


Part of a vote map of an uncompressed image.

Among the  $n = 625$  votes,  
 $k = 56$  voted for (0,0)

8.9% votes

$> 1.56\% = 1/64$



Part of a vote map of a compressed image.

The detection threshold is based on the *non-accidentalness principle*:

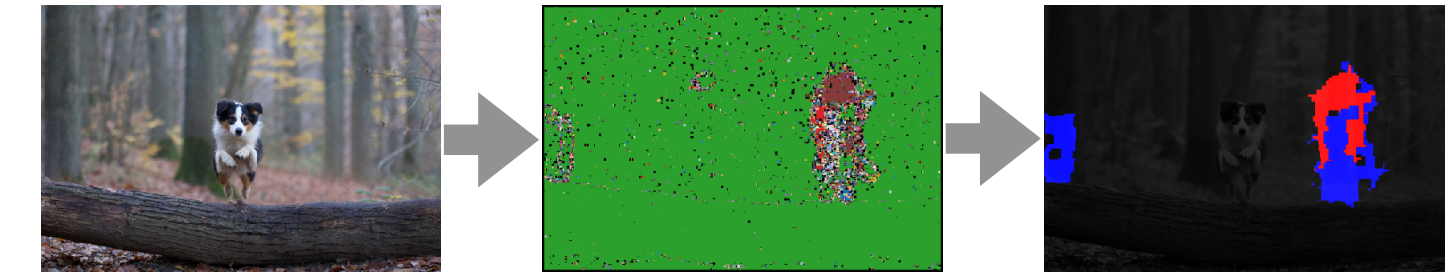
*An observed structure is meaningful only when the relation between its parts is too regular to be the result of an accidental arrangements of independent parts.*

The *a contrario* detection theory is based on a statistical formulation of this *non-accidentalness principle*.

[Wagemans 1992]  
[Albert, Hoffman 1995]  
[Desolneux et al. 2000]  
[Desolneux et al. 2008]



# A *contrario* validation framework

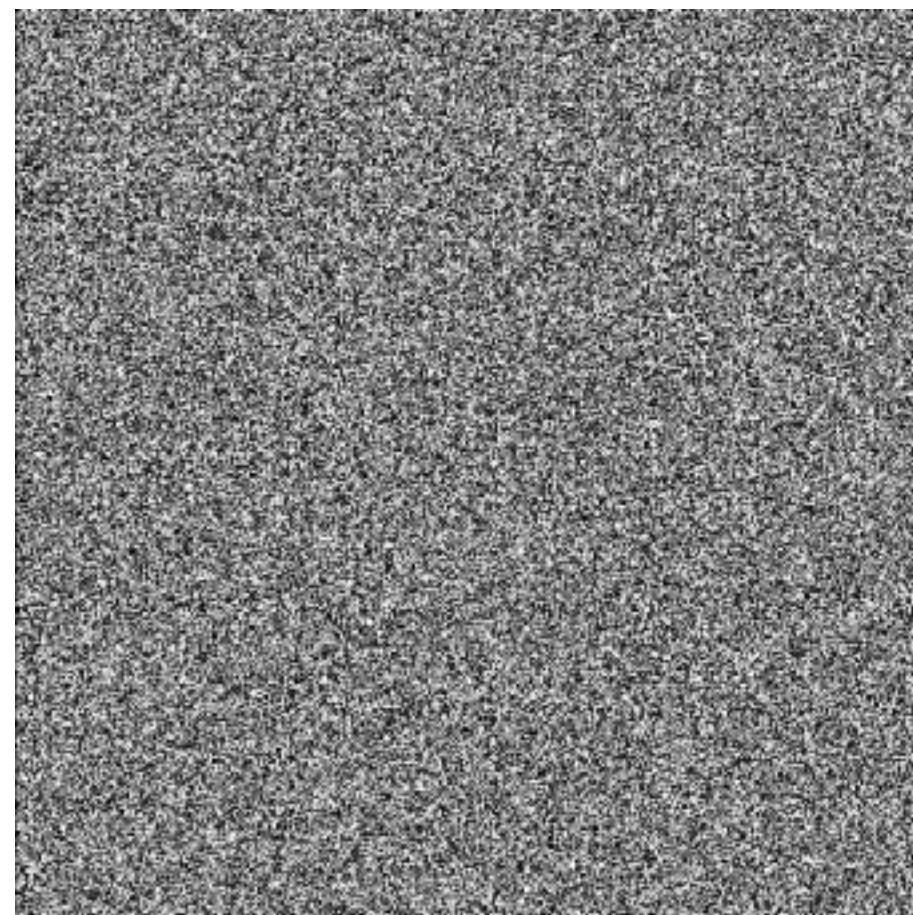


We define for an image of size  $X \times Y$ ,

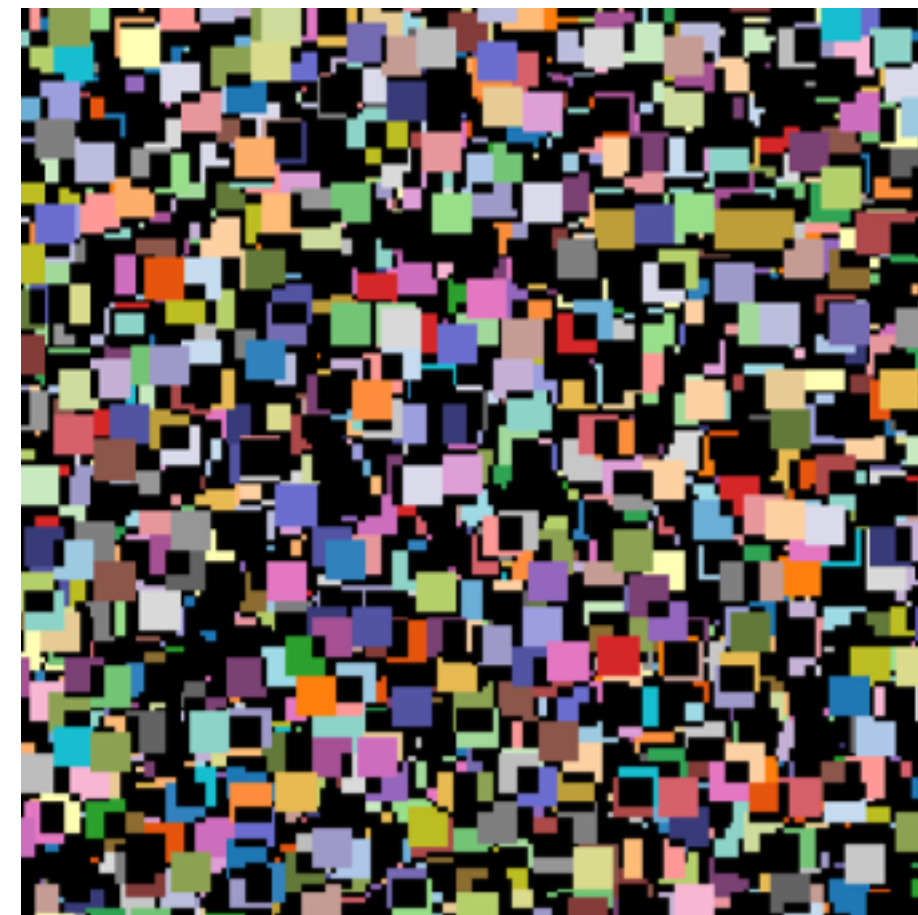
Background model  $H_0$ : the votes at distance larger than 8 are i.i.d.  $\sim U\{1, \dots, 64\}$

Family of tests:

Observation for each test:



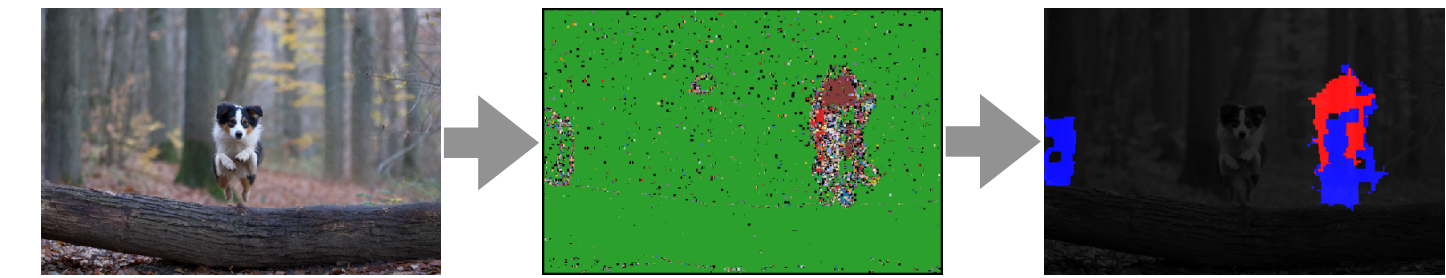
Gaussian noise



Vote map

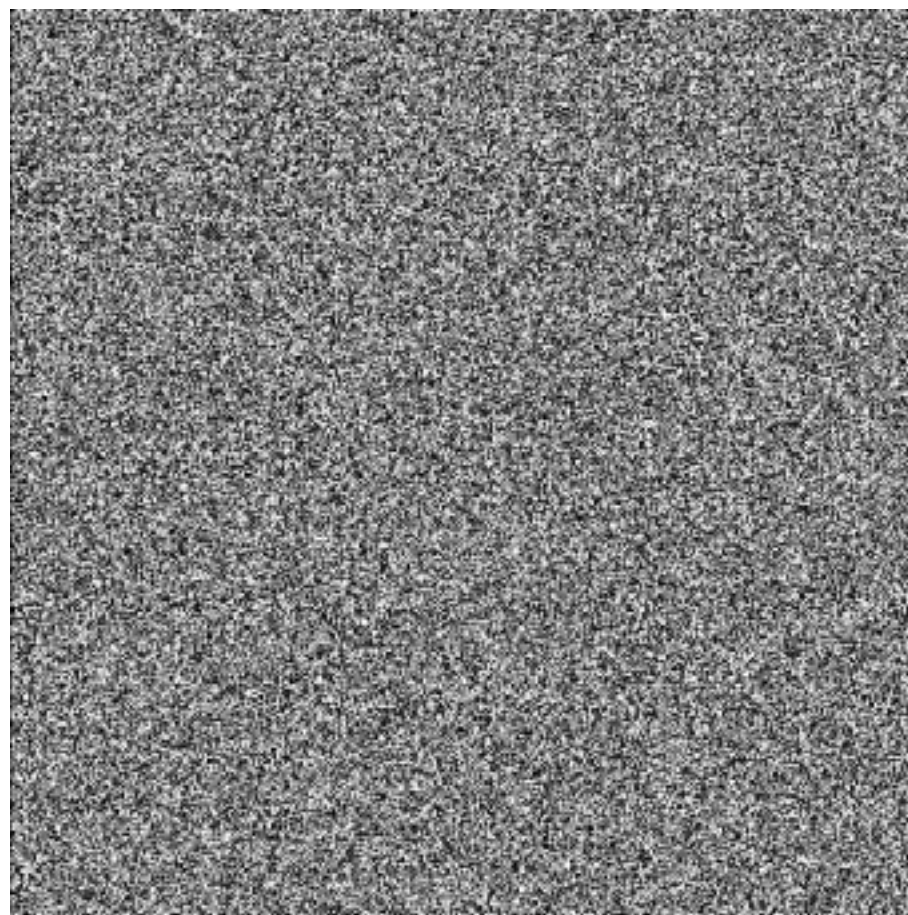
**$H_0$  hypothesis:** votes at distance 8 are independent and uniformly distributed among all the 64 grid origins.

# A *contrario* validation framework

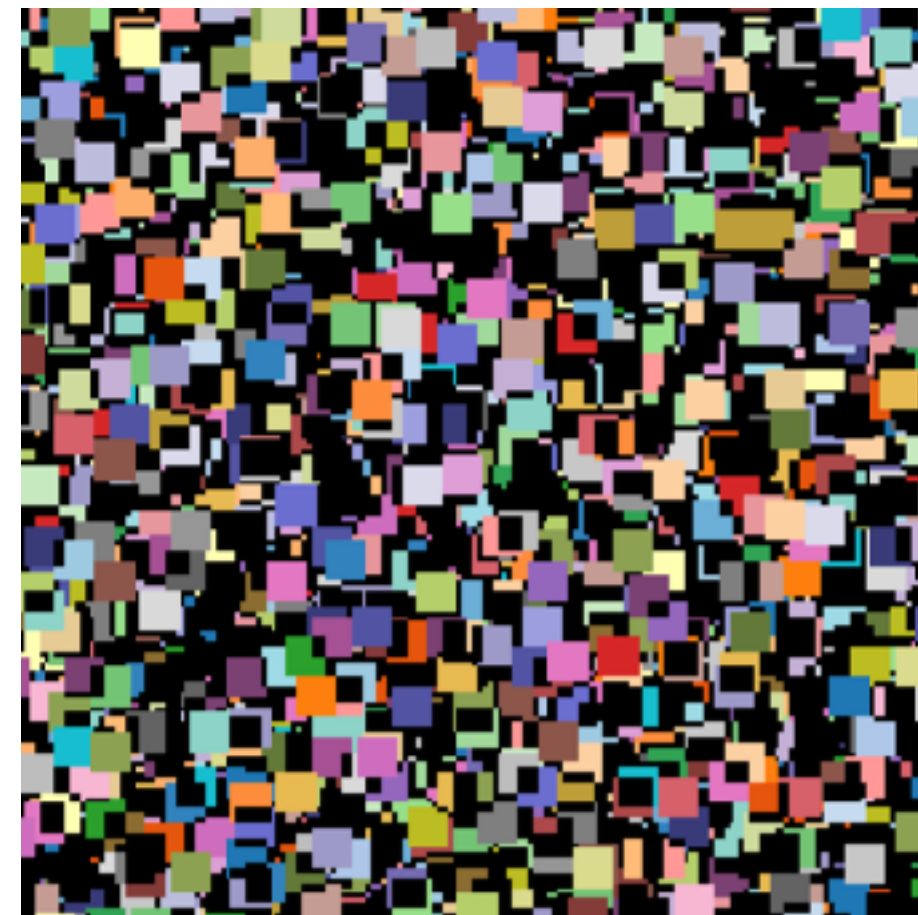


We define for an image of size  $X \times Y$ ,

Background model  $H_0$ : the votes at distance larger than 8 are i.i.d.  $\sim U\{1, \dots, 64\}$   
Family of tests: all the windows in the image and all the 64 candidate grid origins  
Observation for each test:

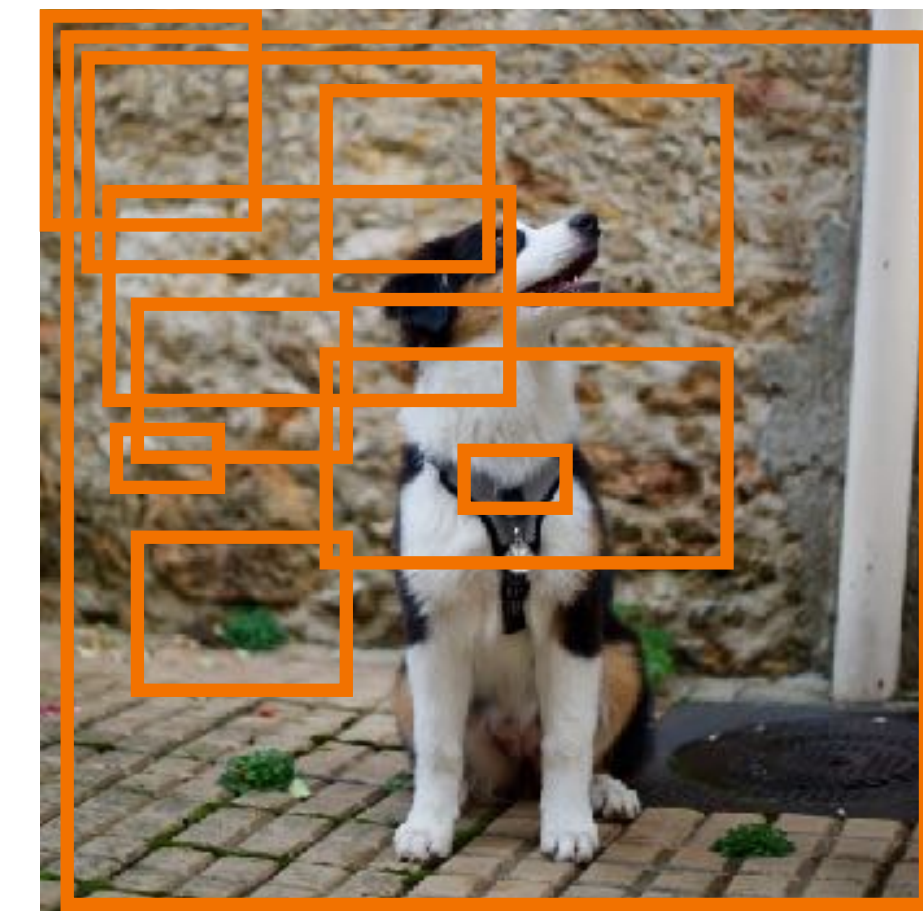


Gaussian noise



Vote map

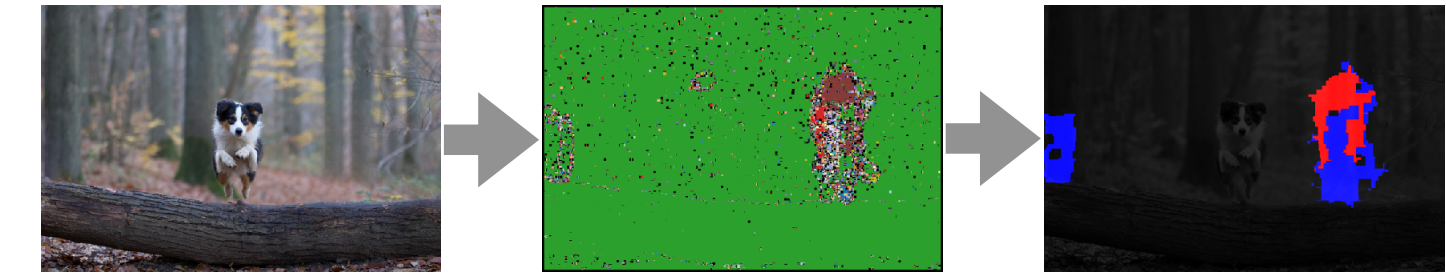
**$H_0$  hypothesis**: votes at distance 8 are independent and uniformly distributed among all the 64 grid origins.



**Family of tests**: all the windows in the image for votes taken at distance 8, and all the 64 different candidates.

$$N_T = 64 \times (XY)^2 \times 64$$

# A *contrario* validation framework



We define for an image of size  $X \times Y$ ,

Background model  $H_0$ : the votes at distance larger than 8 are i.i.d.  $\sim U\{1, \dots, 64\}$   
Family of tests: all the windows in the image and all the 64 candidate grid origins  
Observation for each test: number of votes for a candidate grid origin  $g$

Given a window  $w$ ,  $k$  is the observed number of votes for a candidate grid origin  $g$  among  $n$  votes.

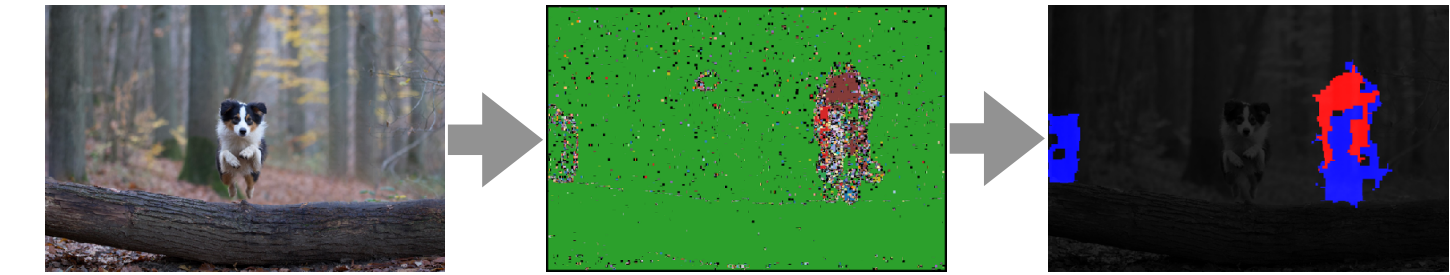
Under  $H_0$ , votes for the grid  $g$ , become a Binomial distributed random variable  $K$  with probability  $p = 1/64$ .

$$P(K \geq k) = \mathcal{B}(n, k, p) = \sum_{j=k}^n \binom{n}{j} p^j (1-p)^{n-j}$$

Probability of obtaining at least  $k$  votes under  $H_0$ .

When this probability is small enough, there exists evidence to reject the  $H_0$  hypothesis and declare that the votes for the grid origin  $g$  are significant.

# A *contrario* validation framework



We define for an image of size  $X \times Y$ ,

Background model  $H_0$ : the votes at distance larger than 8 are i.i.d.  $\sim U\{1, \dots, 64\}$   
Family of tests: all the windows in the image and all the 64 candidate grid origins  
Observation for each test: number of votes for a candidate grid origin  $g$

If the hypothesis is rejected, then the detection is significant.

Given a window  $w$ ,  $k$  is the observed number of votes for a candidate grid origin  $g$  among  $n$  votes.

Under  $H_0$ , votes for the grid  $g$ , become a Binomial distributed random variable  $K$  with probability  $p = 1/64$ .

$$P(K \geq k) = \mathcal{B}(n, k, p) = \sum_{j=k}^n \binom{n}{j} p^j (1-p)^{n-j}$$

Probability of obtaining at least  $k$  votes under  $H_0$ .

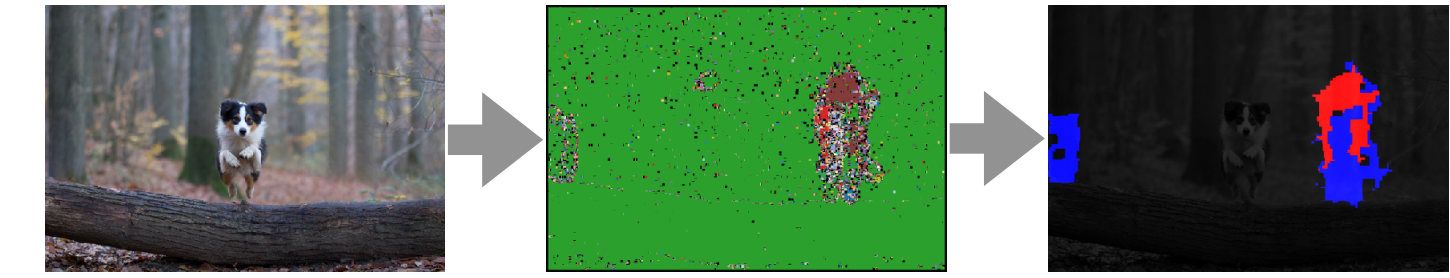
When this probability is small enough, there exists evidence to reject the  $H_0$  hypothesis and declare that the votes for the grid origin  $g$  are significant.

Following the *a contrario* methodology, we define the Number of False Alarms (**NFA**) of a candidate grid  $g$  on a given window  $w$  as

$$\text{NFA}(g, w) = N_T P(K \geq k)$$

We compute the **NFA** for all the candidates  $g$  and we say that a JPEG grid is detected when at least one is  $< \varepsilon$ .

# A *contrario* validation framework



We define for an image of size  $X \times Y$ ,

Background model  $H_0$ : the votes at distance larger than 8 are i.i.d.  $\sim U\{1, \dots, 64\}$   
Family of tests: all the windows in the image and all the 64 candidate grid origins  
Observation for each test: number of votes for a candidate grid origin  $g$

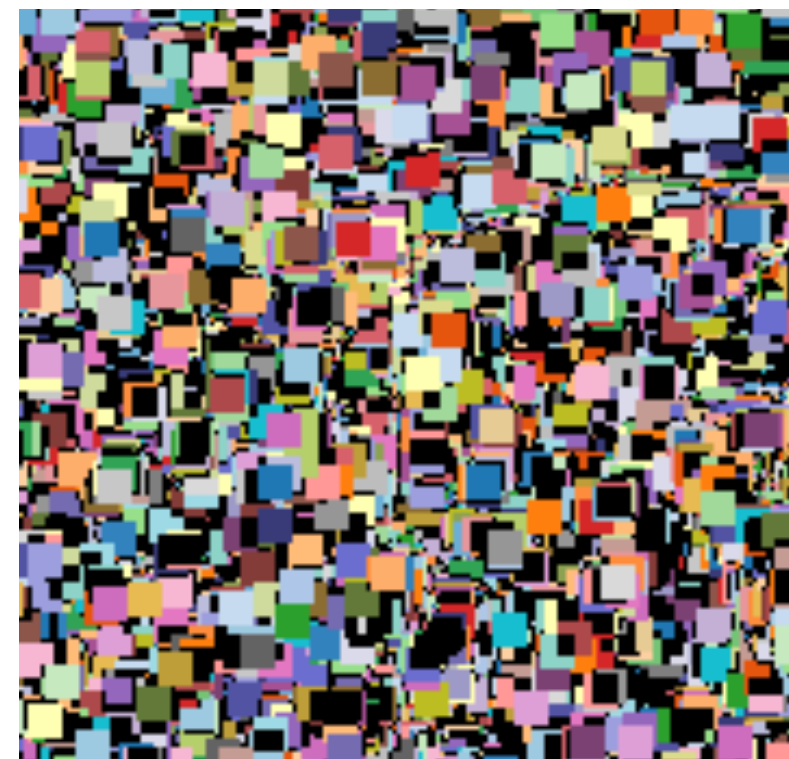
If the hypothesis is rejected, then the detection is significant.  $\varepsilon = 1$

Among the  $n = 625$  votes,  
 $k = 30$  voted for  $(0,0)$

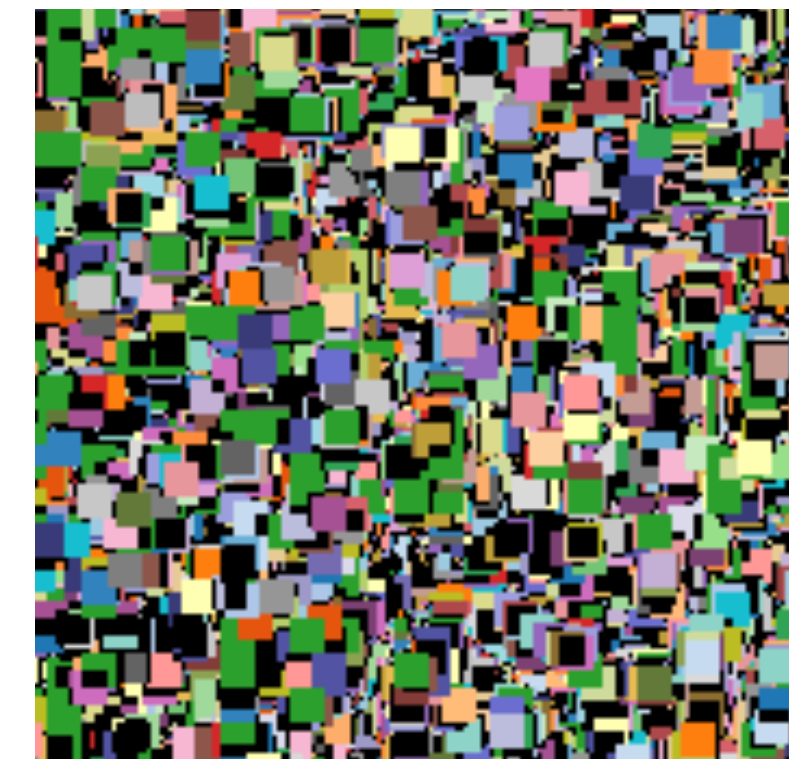
$$\text{NFA}((0,0), w) = 10^{5.84598} > 1$$

$$\forall g \in \{1, \dots, 63\}, \text{NFA}(g, w) > 1$$

**No JPEG grid detected**



Part of a vote map of an uncompressed image.



Part of a vote map of a compressed image.

Among the  $n = 625$  votes,  
 $k = 56$  voted for  $(0,0)$

$$\text{NFA}((0,0), w) = 10^{-11.5489} < 1$$

$$\forall g \in \{1, \dots, 63\}, \text{NFA}(g, w) > 1$$

**JPEG grid origin  $(0,0)$**

$$\text{NFA}(g, w) = 64^2 \cdot (XY)^2 \cdot \mathcal{B} \left( n, k, \frac{1}{64} \right)$$

# ZERO applied to the whole image

Uncompressed



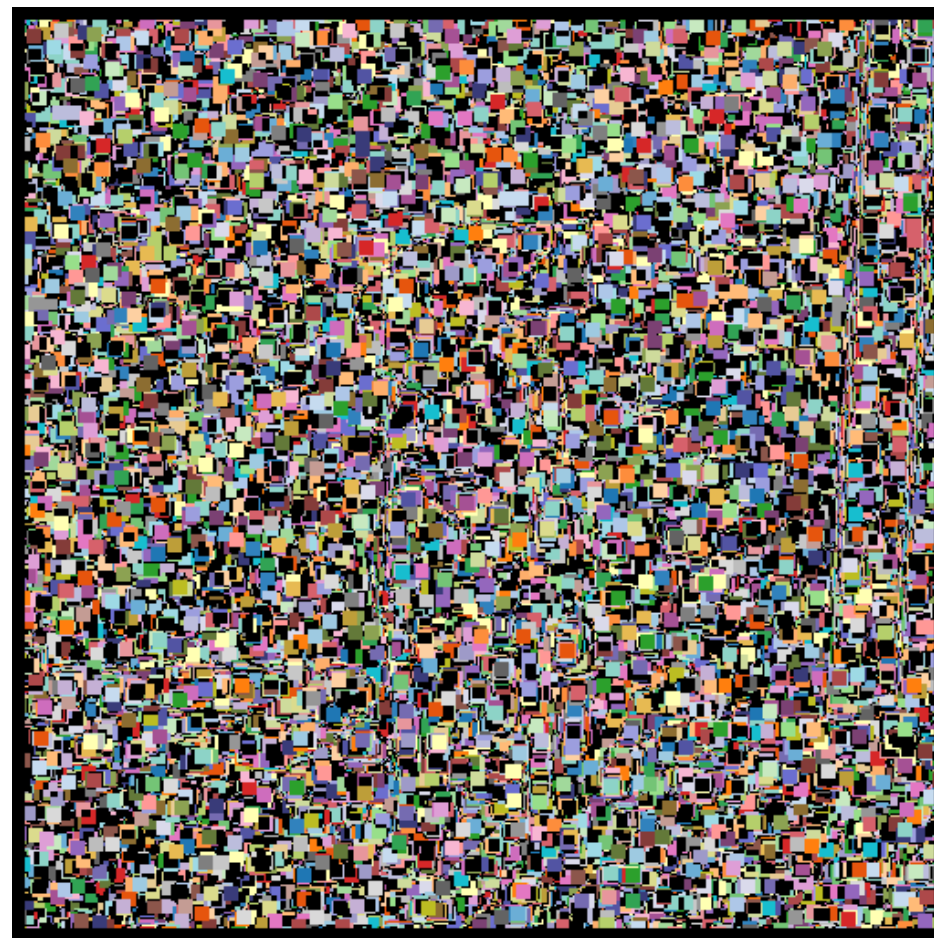
Compressed (QF = 80)



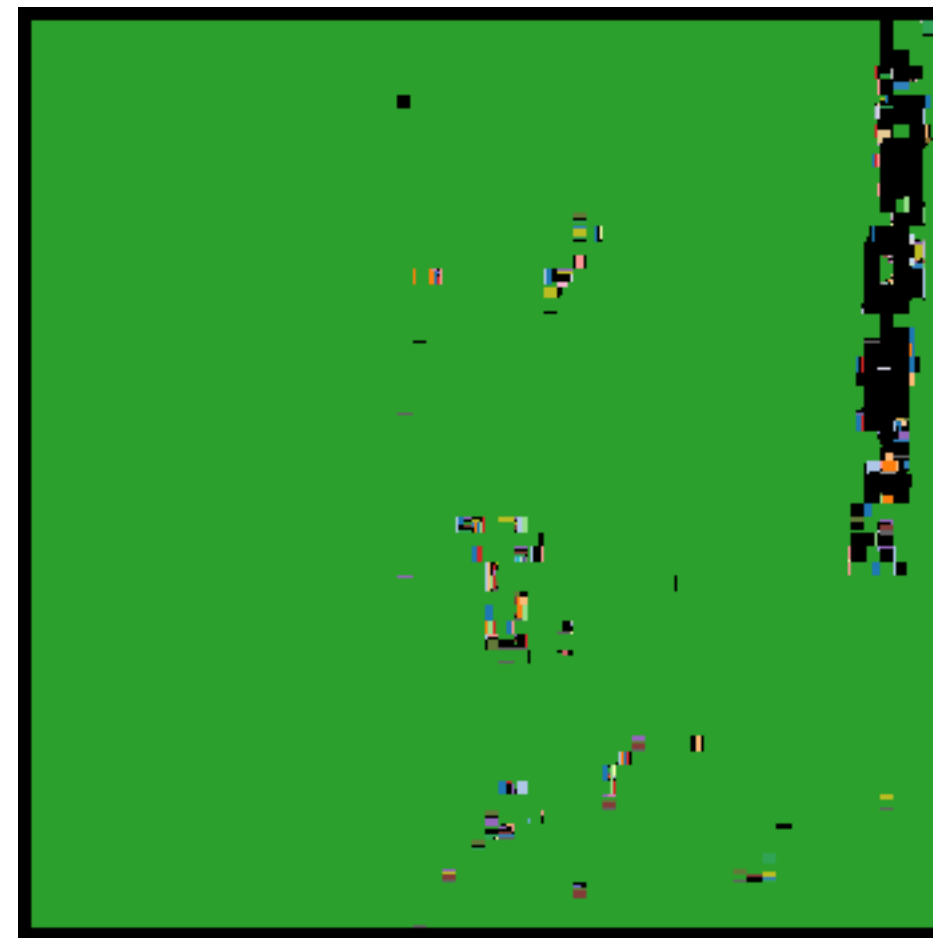
Compressed (QF = 98)



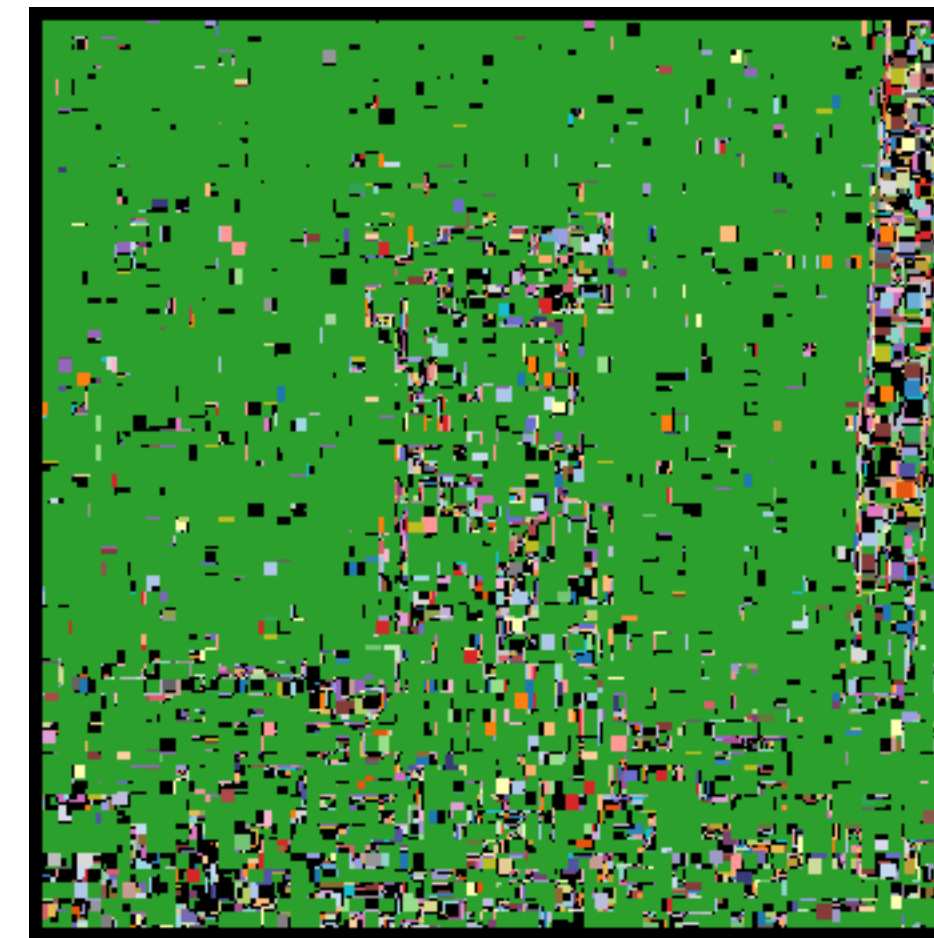
Compressed (QF = 99)



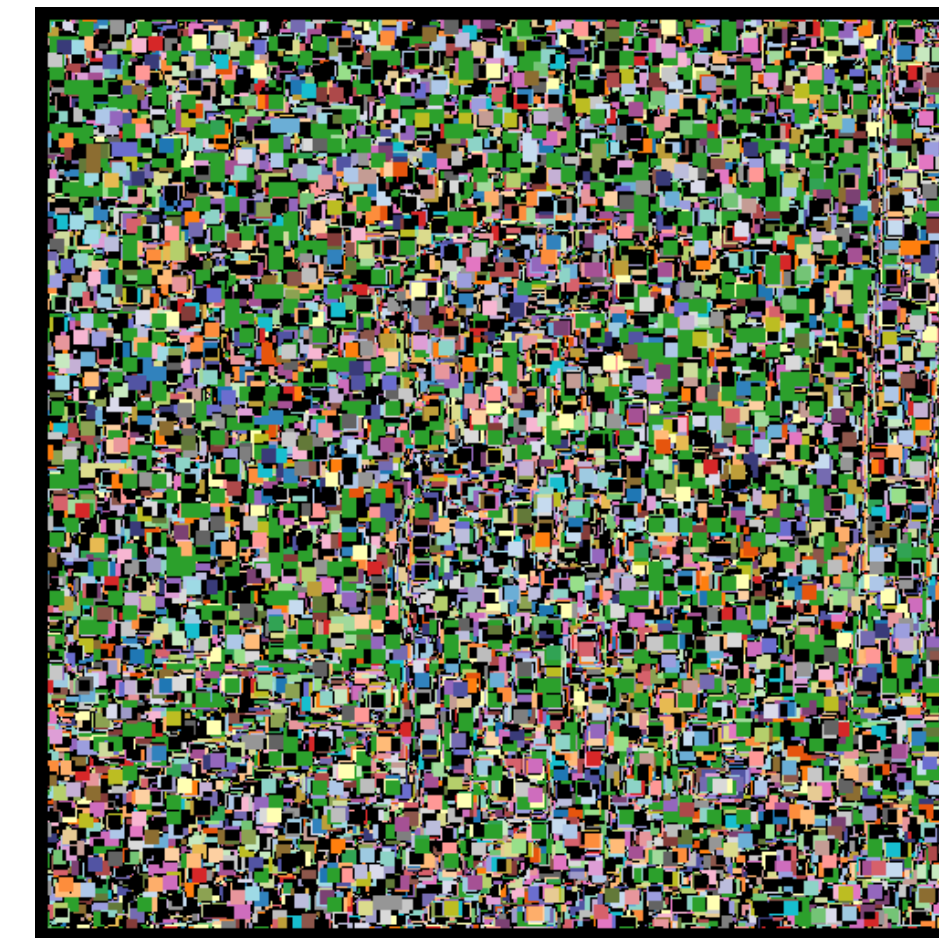
No JPEG grid detected



JPEG grid origin (0,0)



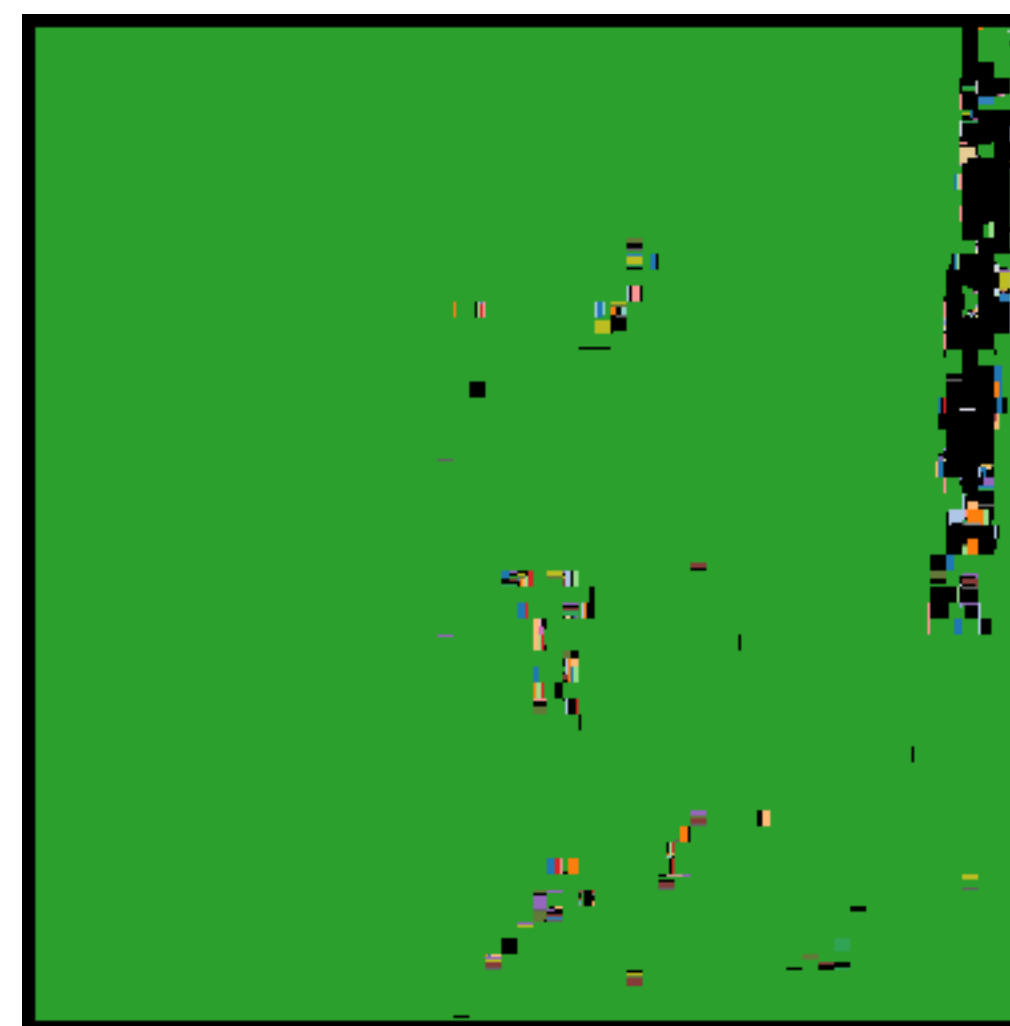
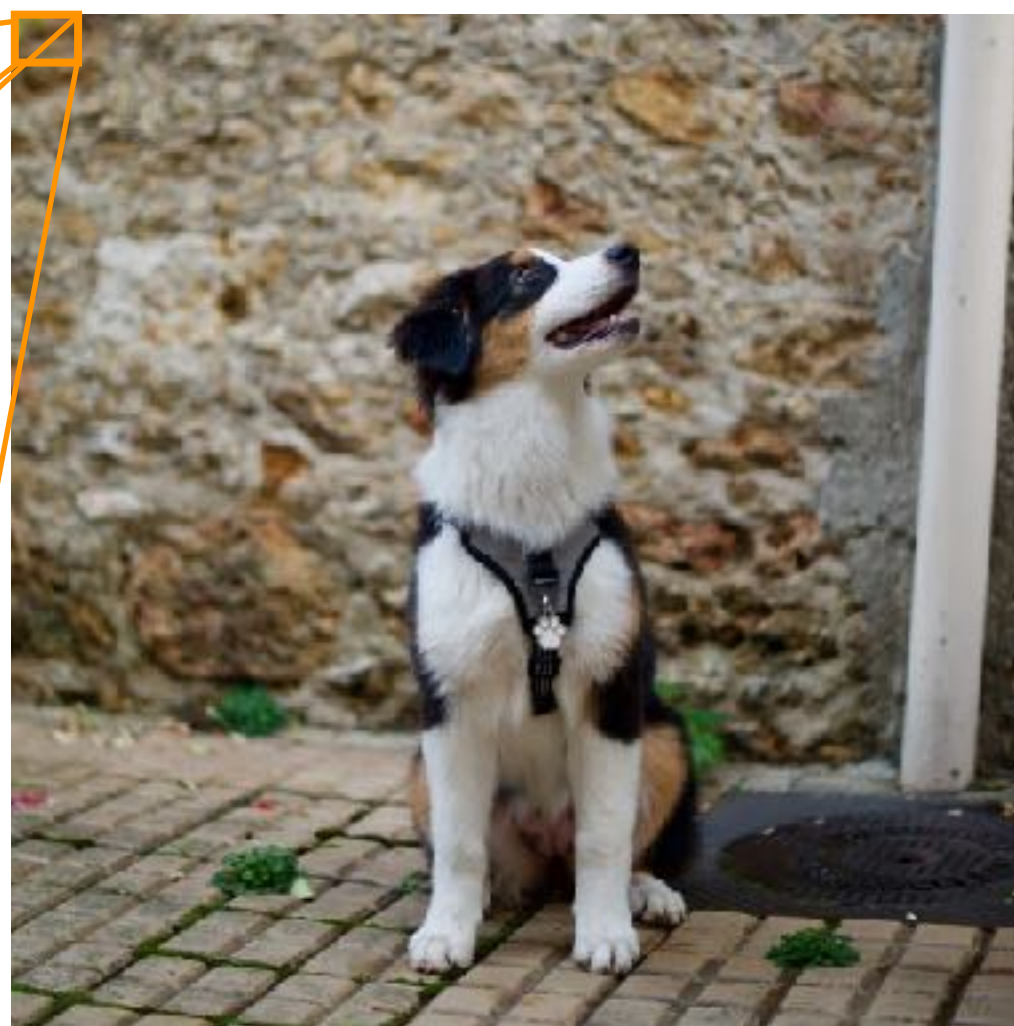
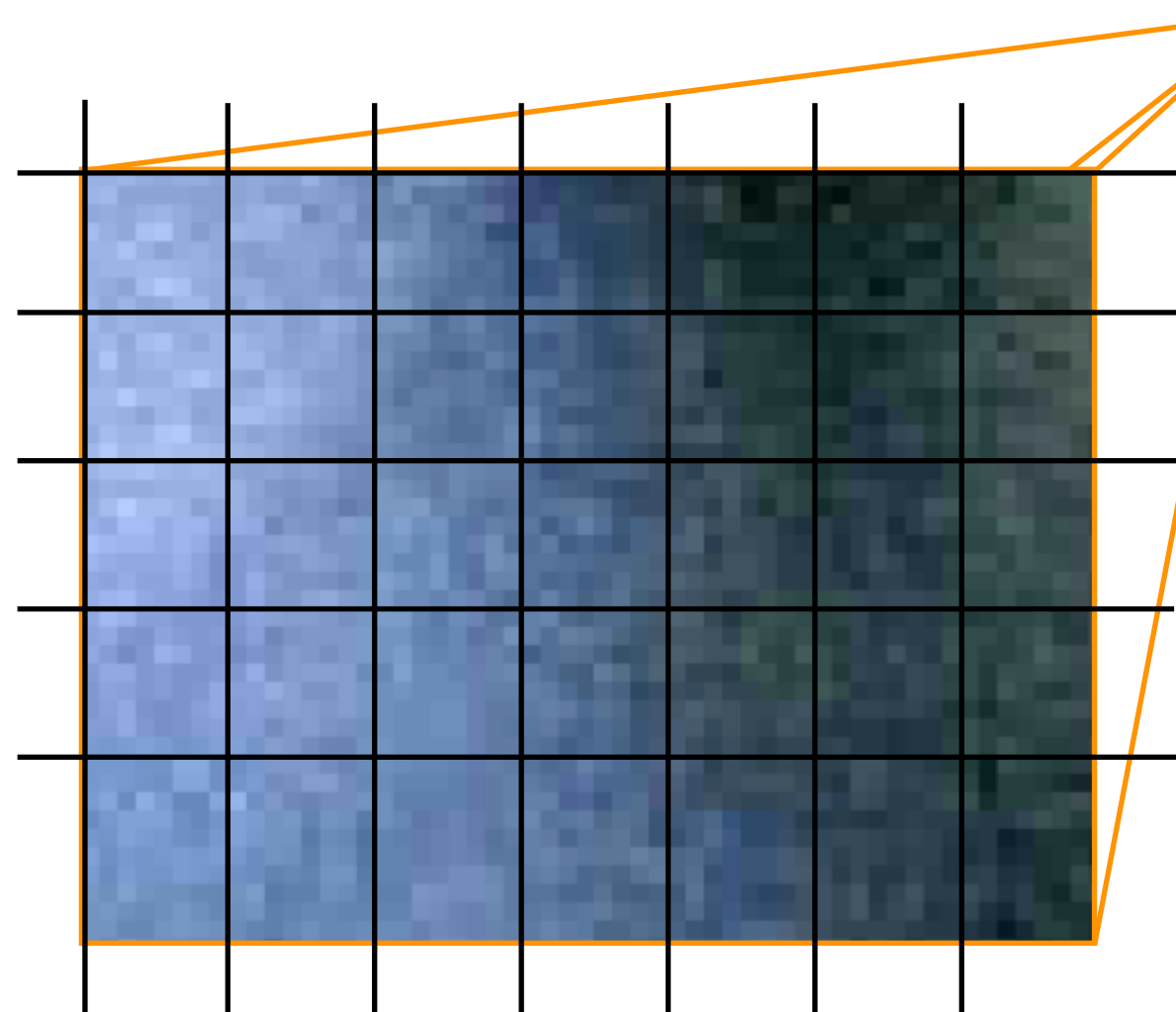
JPEG grid origin (0,0)



JPEG grid origin (0,0)

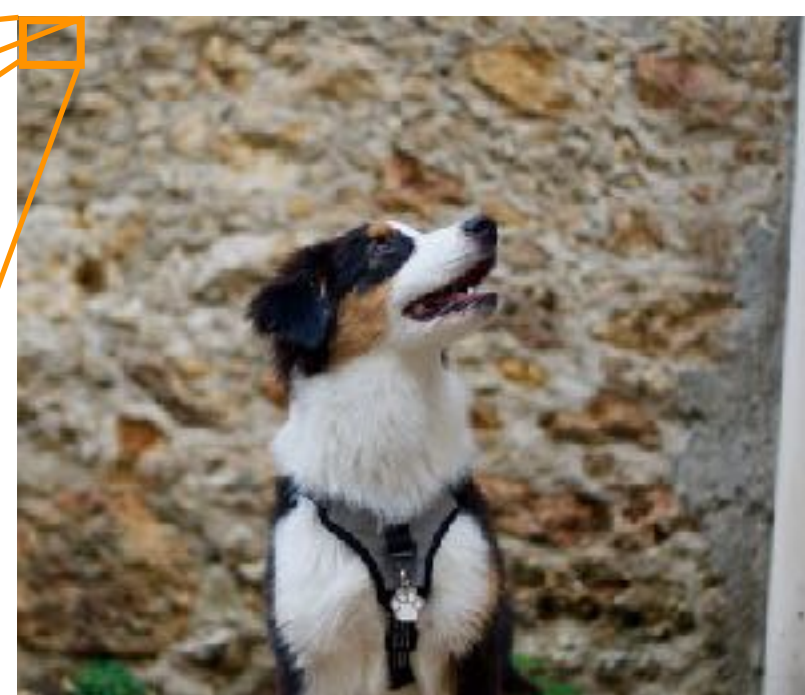
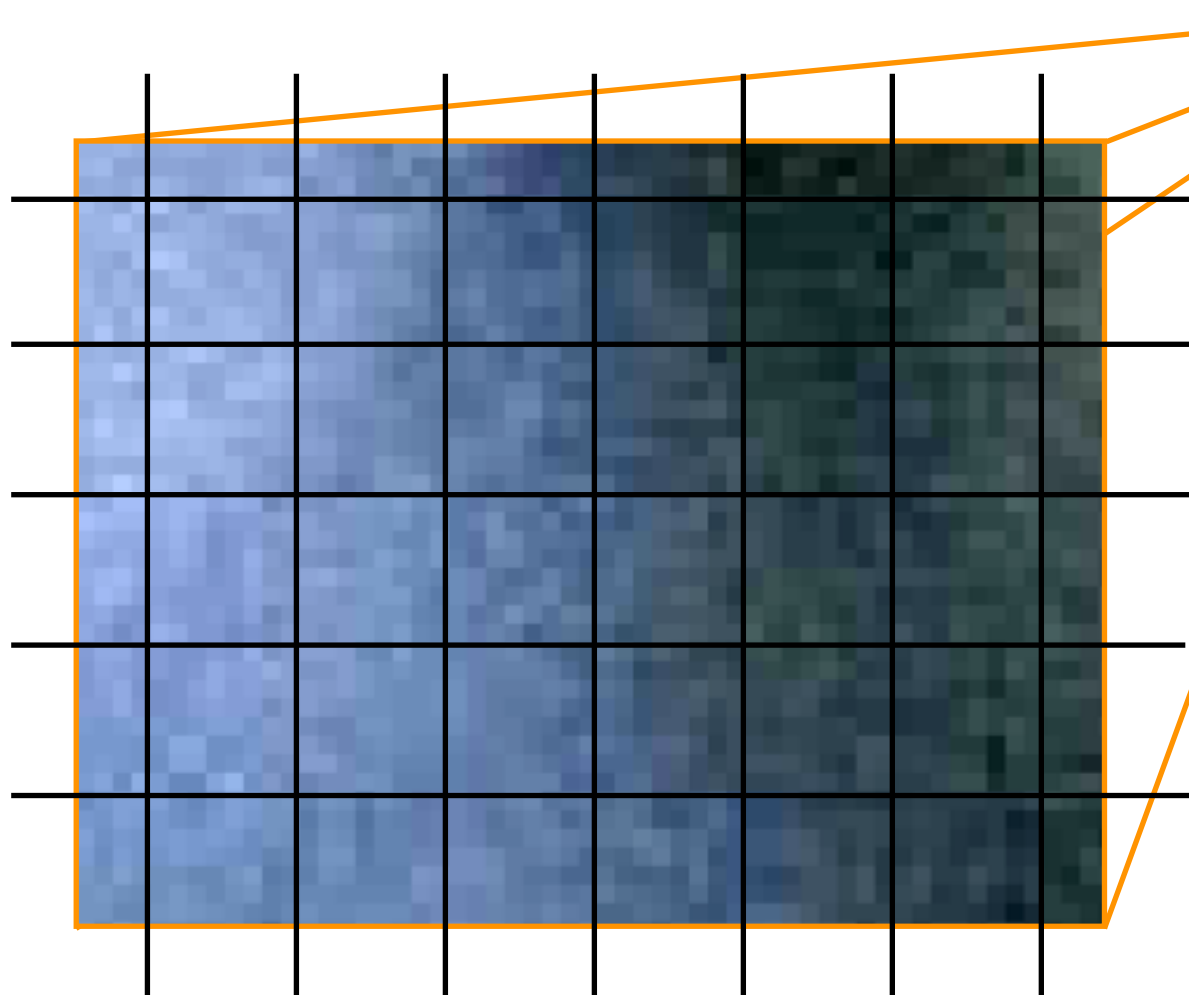
# ZERO: a crop detector

Main grid (0,0)



JPEG grid origin (0,0)

Main grid (6,6)

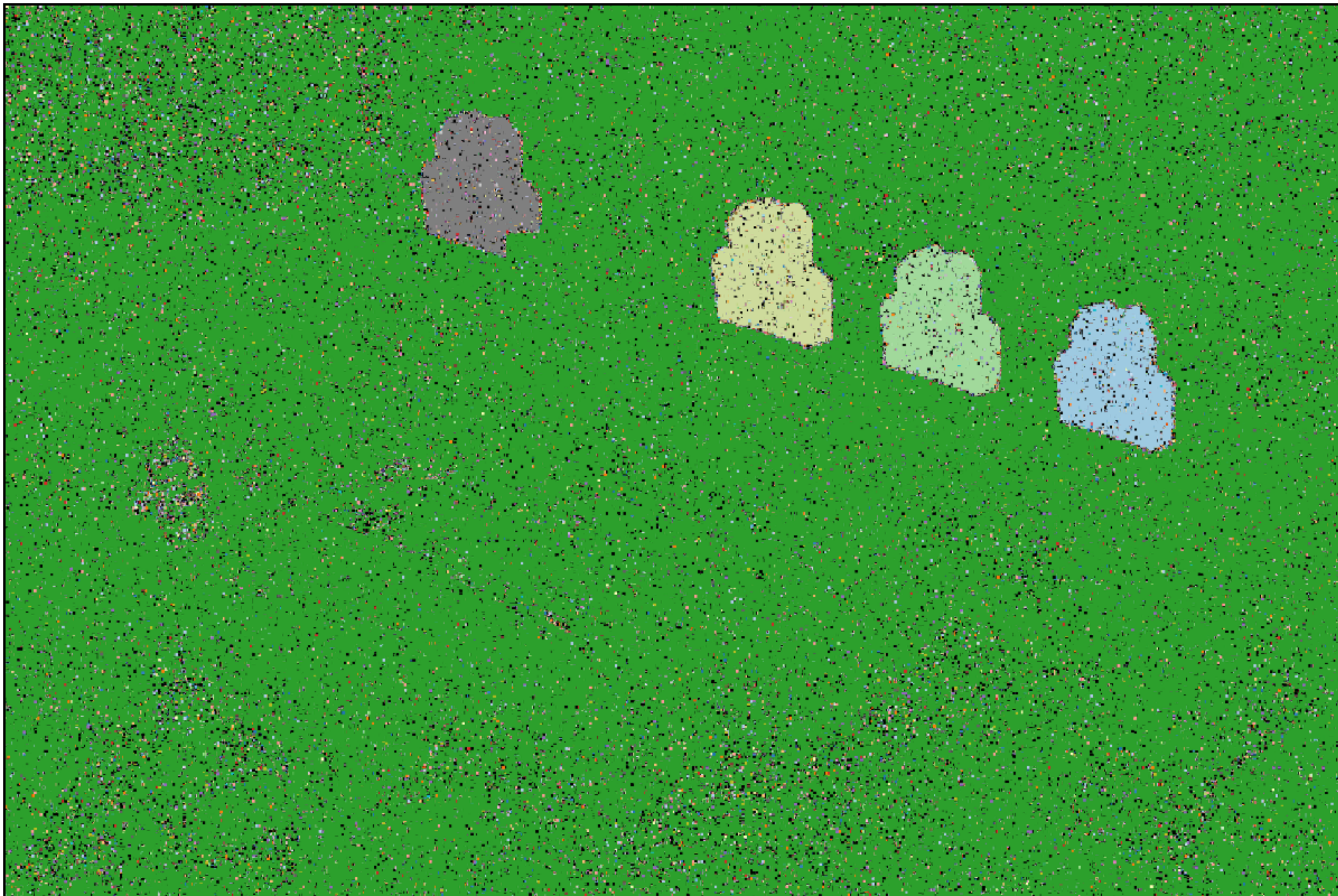


JPEG grid origin (6,6)

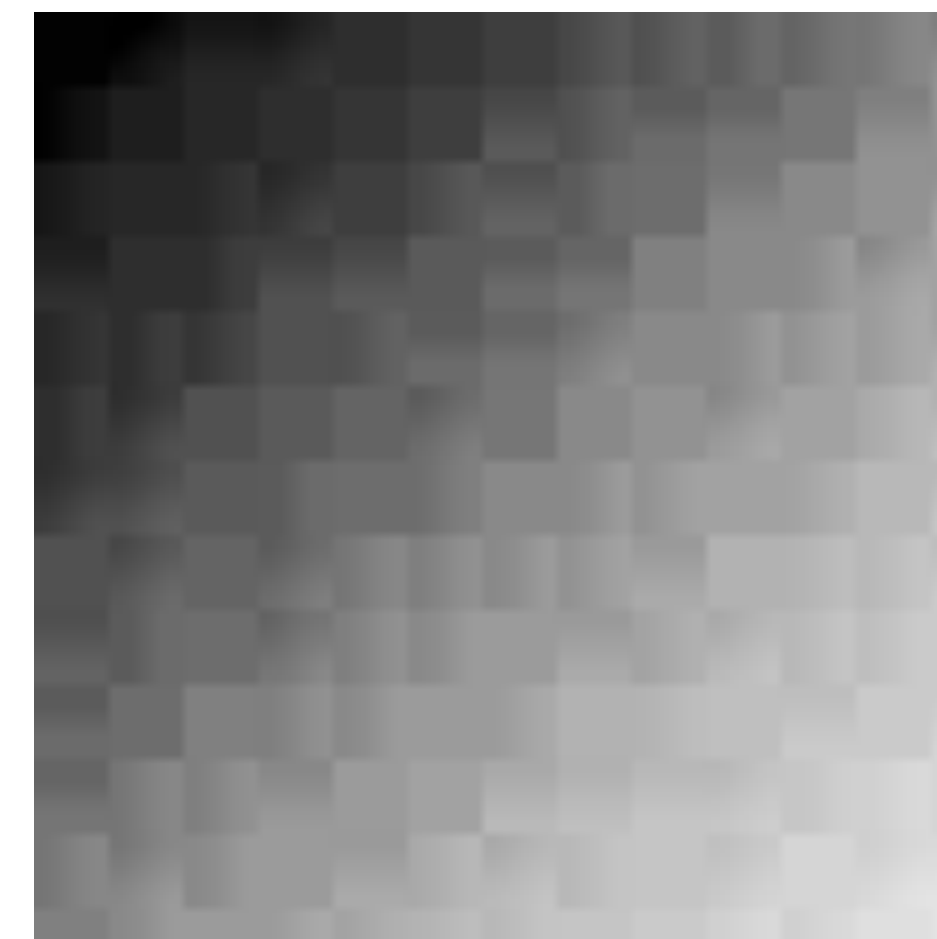
**How to detect forgeries?**



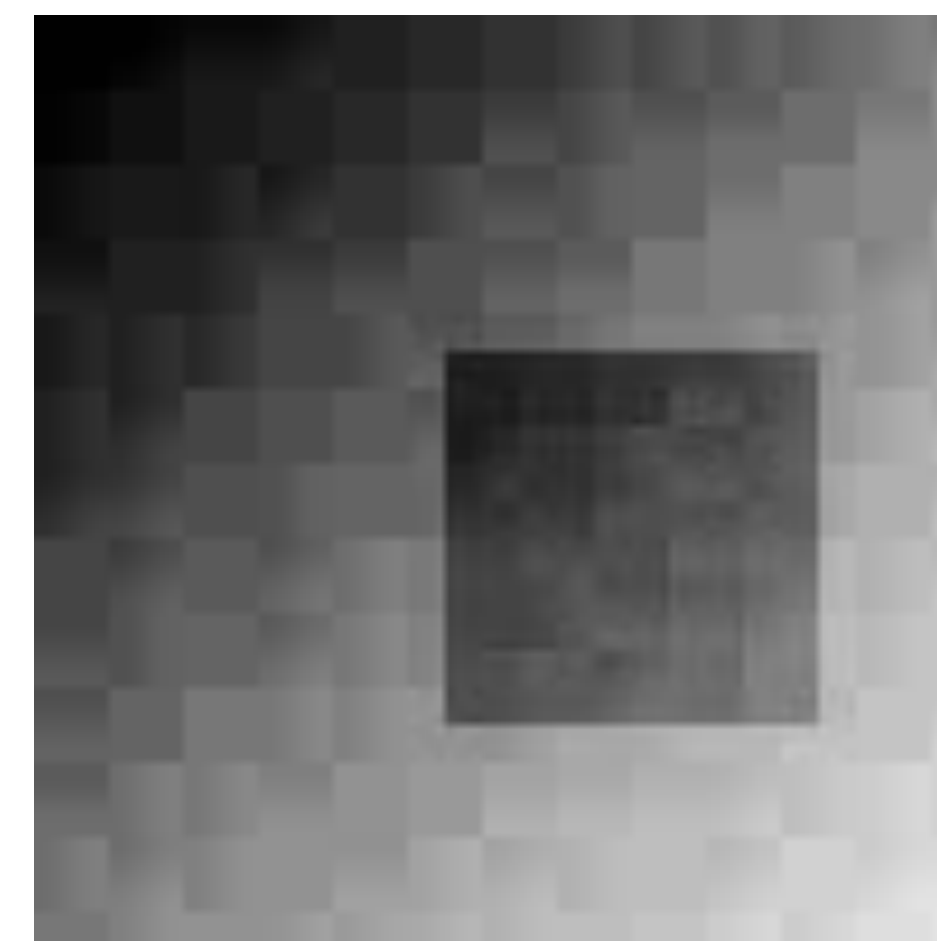
# ZERO: a forgery detector



Vote map. JPEG grid origin (0,0).

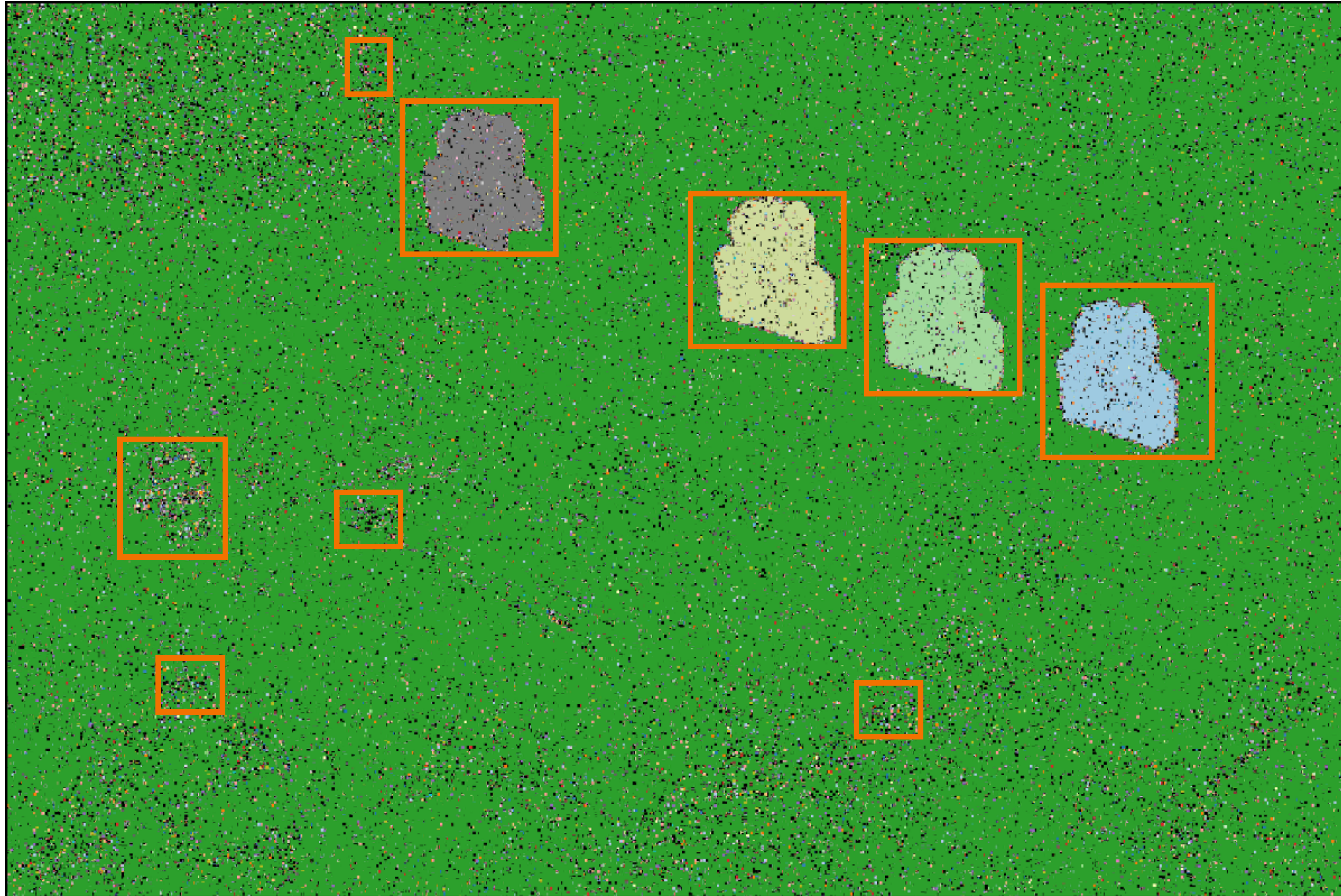


Authentic



Local shifted grid.

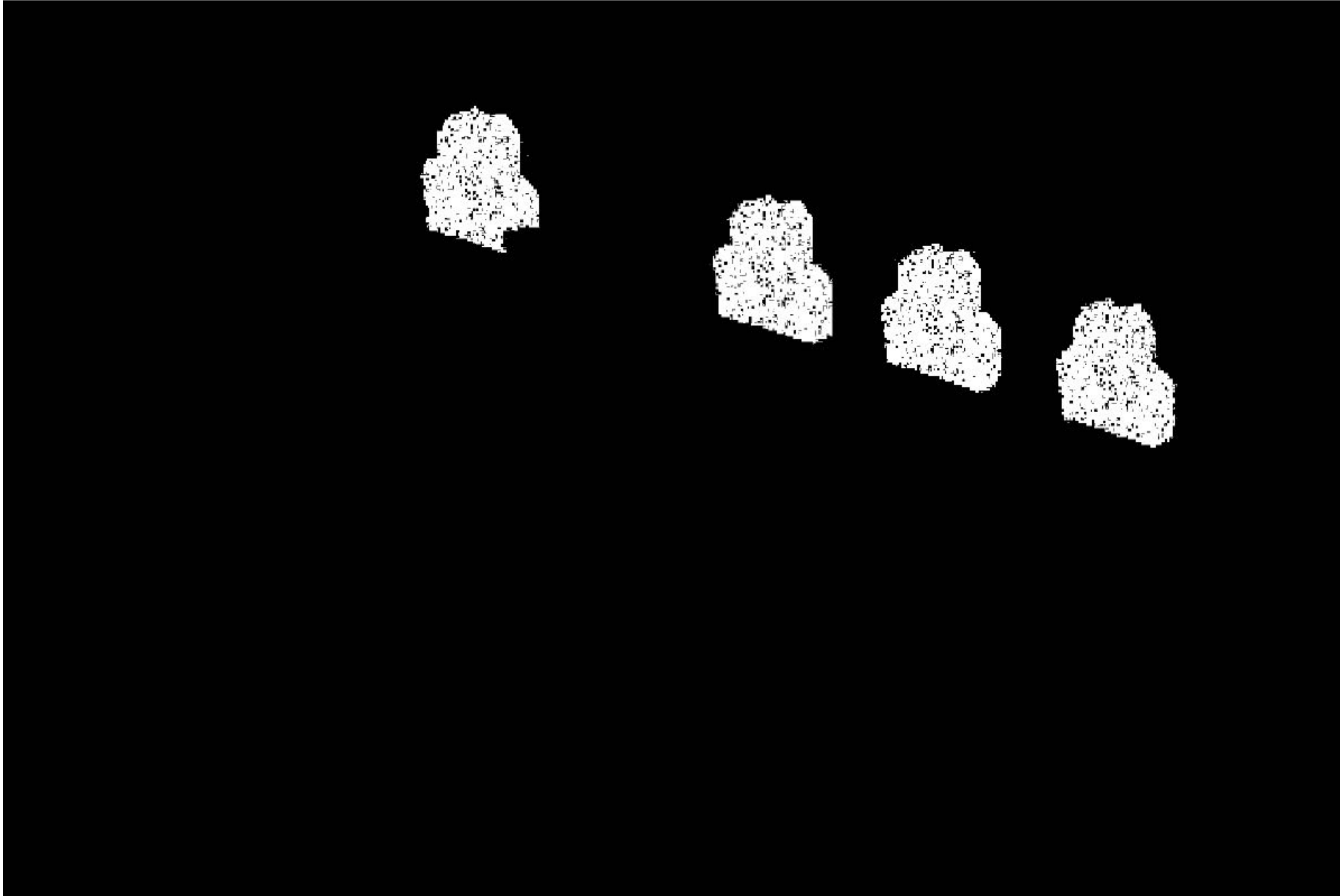
# ZERO: a forgery detector



- Compute the **vote map**
- Detect global grids and **main grid**
- Partition the image into groups of connected pixels which vote alike and different from the main grid
- Create bounding boxes

Bounding boxes around grouping of pixels which did not vote for the main grid.

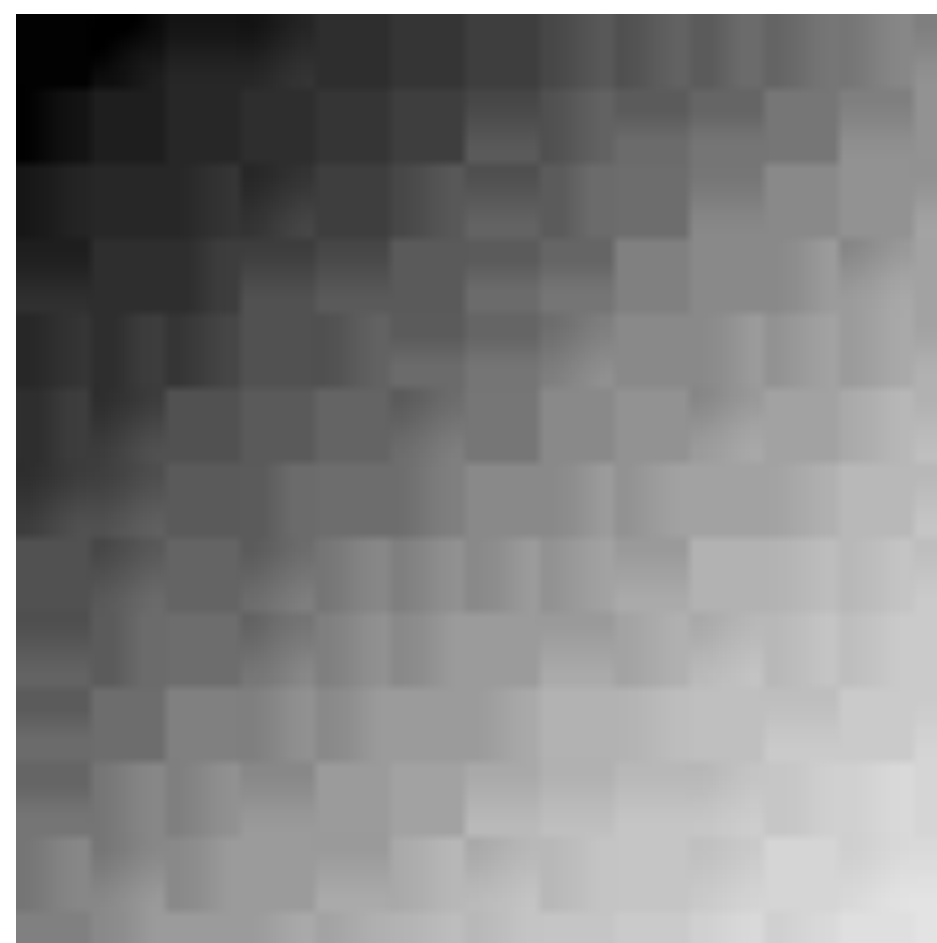
# ZERO: a forgery detector



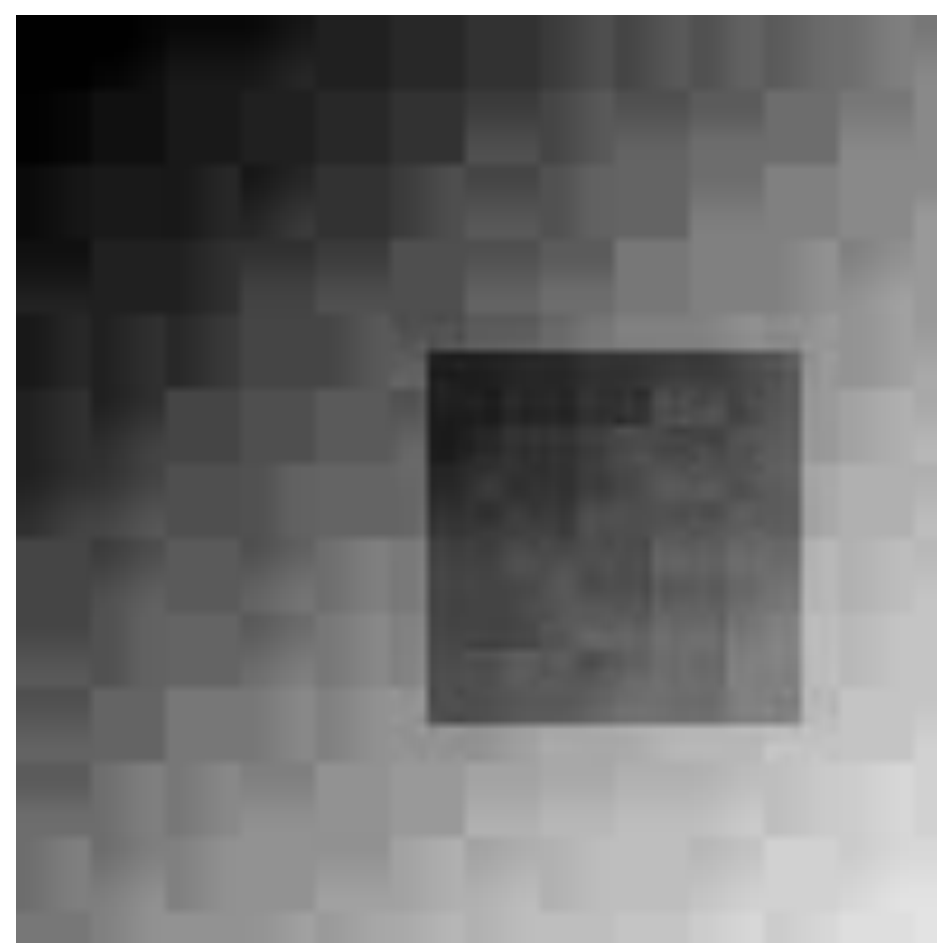
If the window's vote is significant, then the pixels are marked.

- Compute the **vote map**
- Detect global grids and **main grid**
- Partition the image into groups of connected pixels which vote alike and different from the main grid
- Create bounding boxes
- Apply the *a contrario* validation
- Compute **forgery map**

# ZERO: a local foreign grid origin detector



Authentic

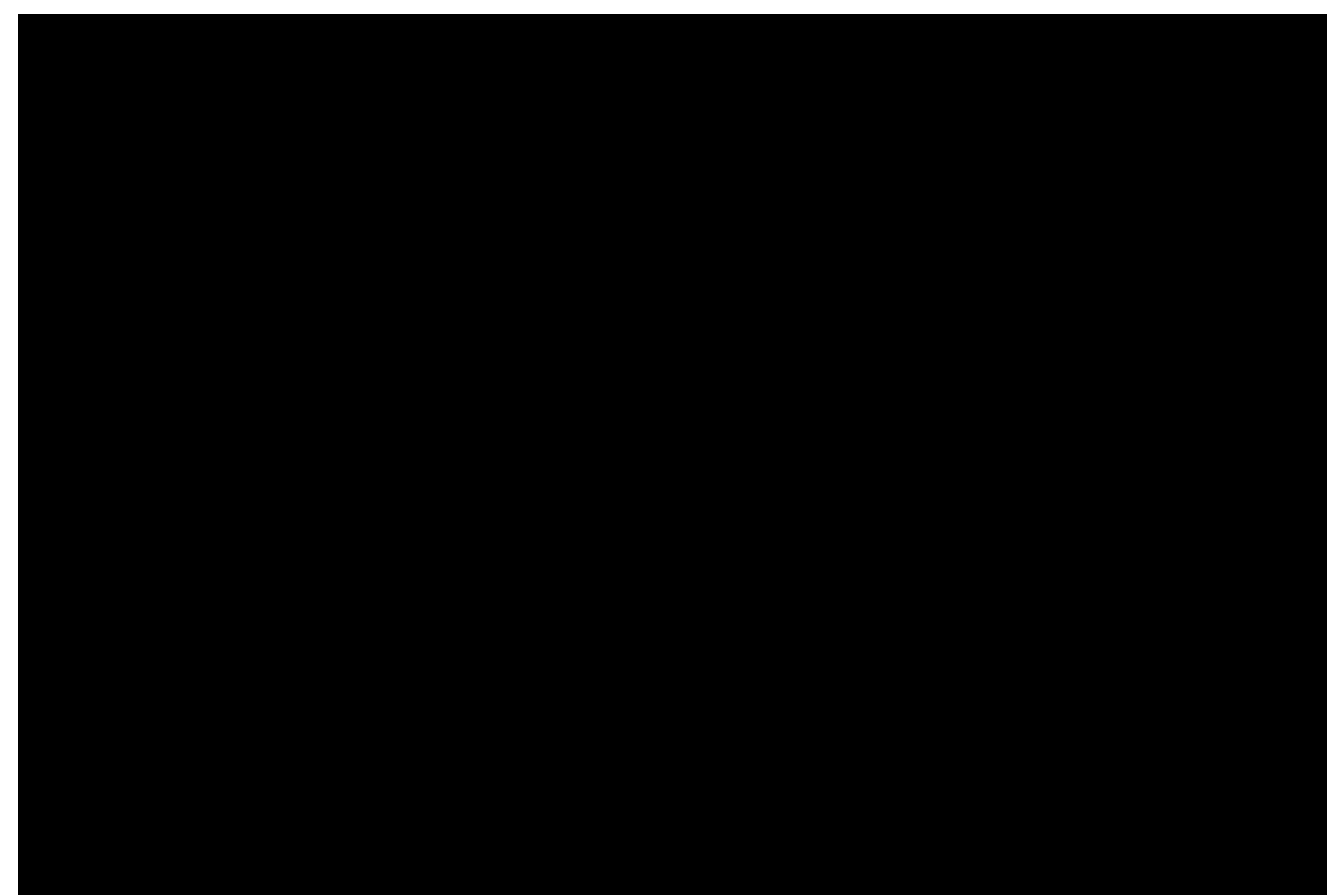


Local shifted grid.

Original

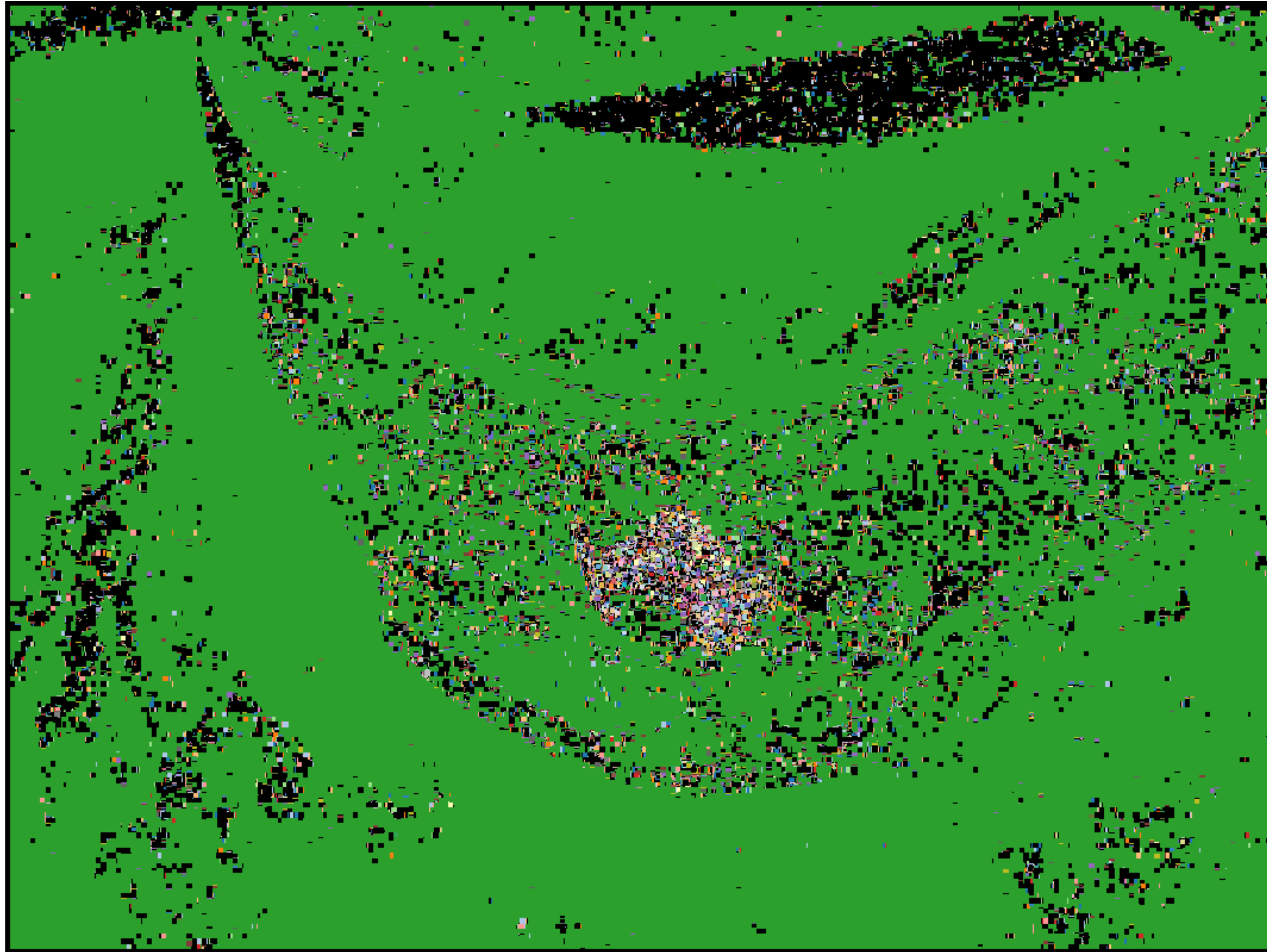


Forged

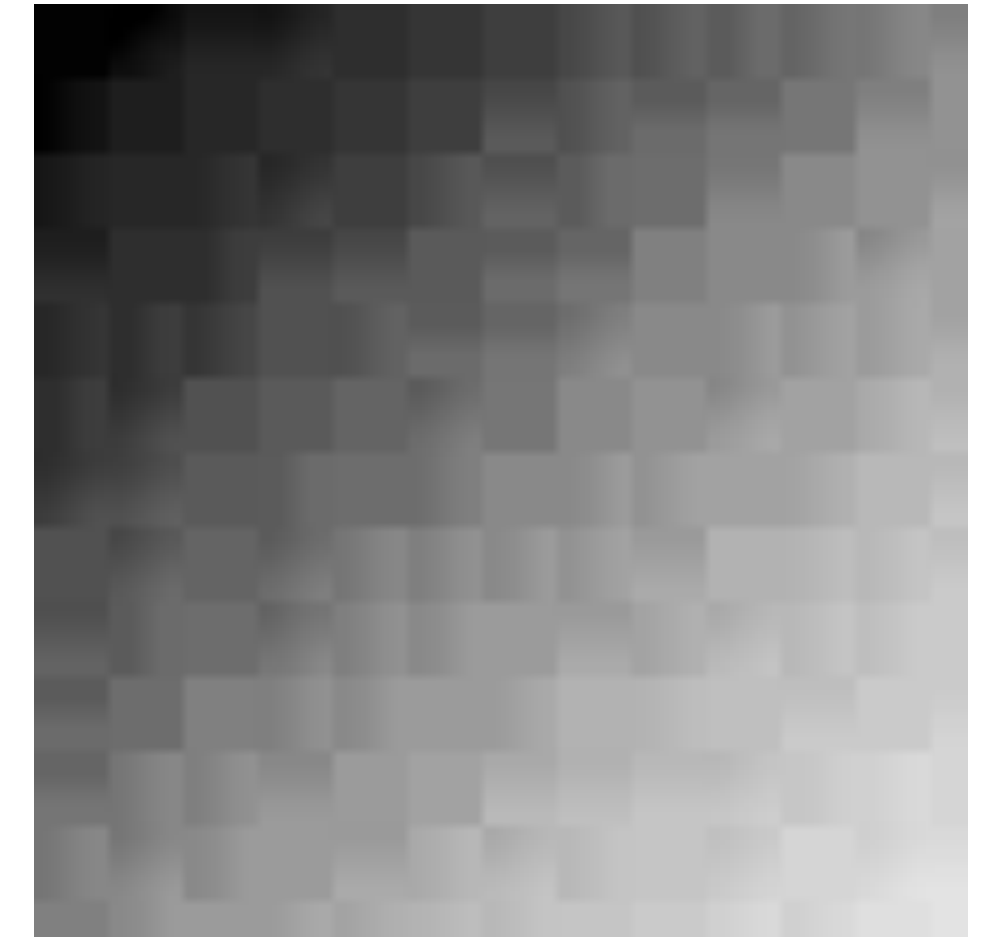


Detection results of images from FAU dataset.

# ZERO: a forgery detector



Vote map. JPEG grid origin (0,0).

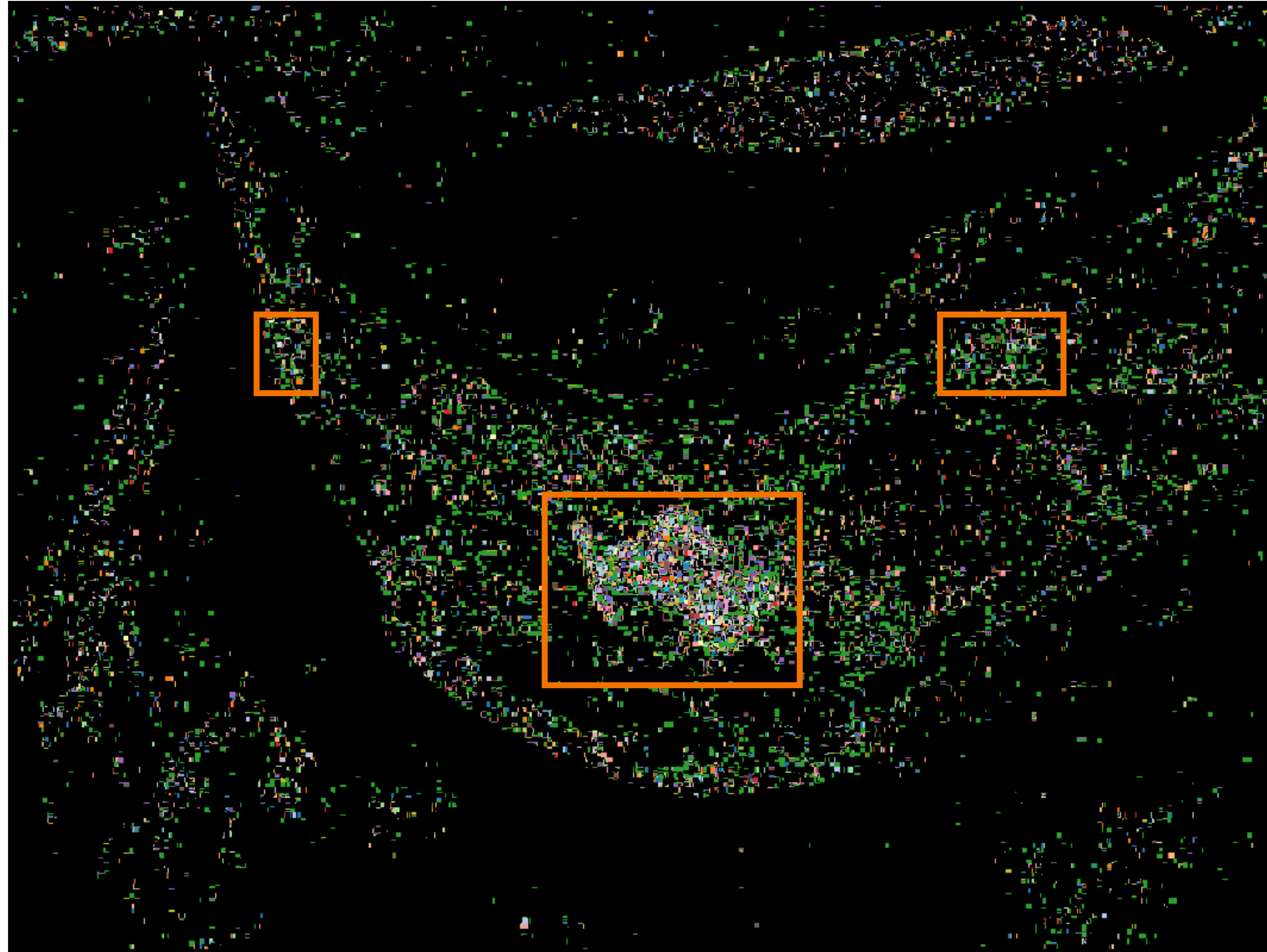


Authentic



Local missing grid.

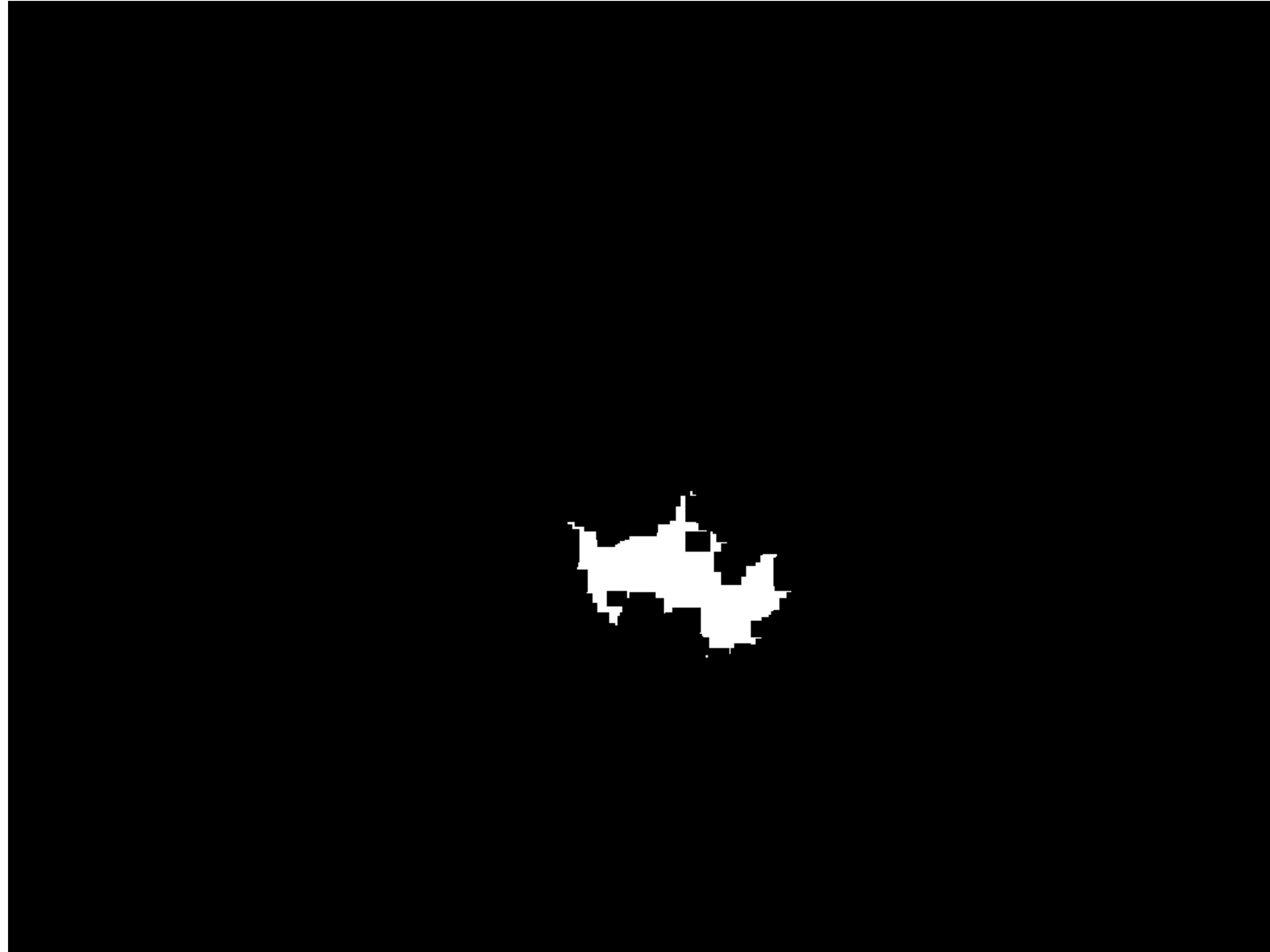
# ZERO: a forgery detector



- Compute the **vote map**
- Detect global grids and **main grid**
- Create compressed version of the image (QF = 99)
- Compute the **new vote map** where the pixels which voted for the main grid are removed
- Partition the image into groups of connected pixels which vote for (0,0)
- Create bounding boxes

New vote map. Pixels which voted for grid (0,0) before do not vote again.

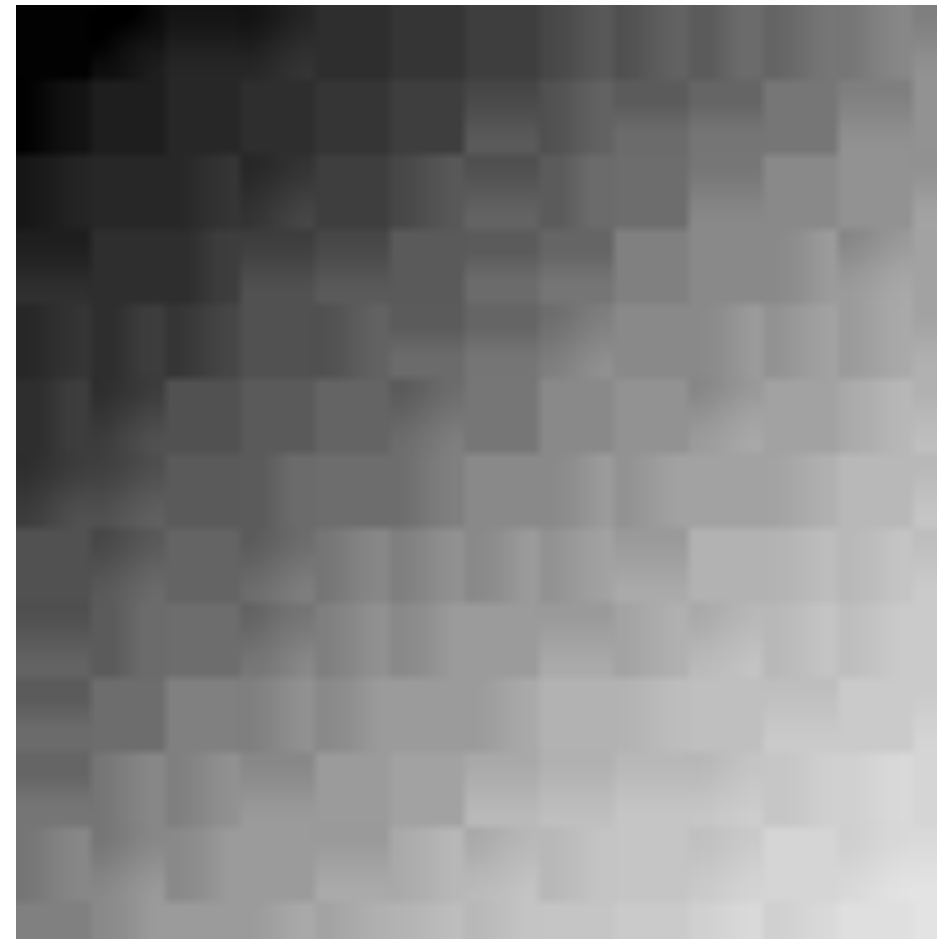
# ZERO: a forgery detector



If the window's vote for (0,0) is significant, then the pixels are marked.

- Compute the **vote map**
- Detect global grids and **main grid**
- Create compressed version of the image (QF = 99)
- Compute the **new vote map** where the pixels which voted for the main grid are removed
- Partition the image into groups of connected pixels which vote for (0,0)
- Create bounding boxes
- Apply the *a contrario* validation
- Compute **forgery map**

# ZERO: a local missing grid origin detector



Authentic

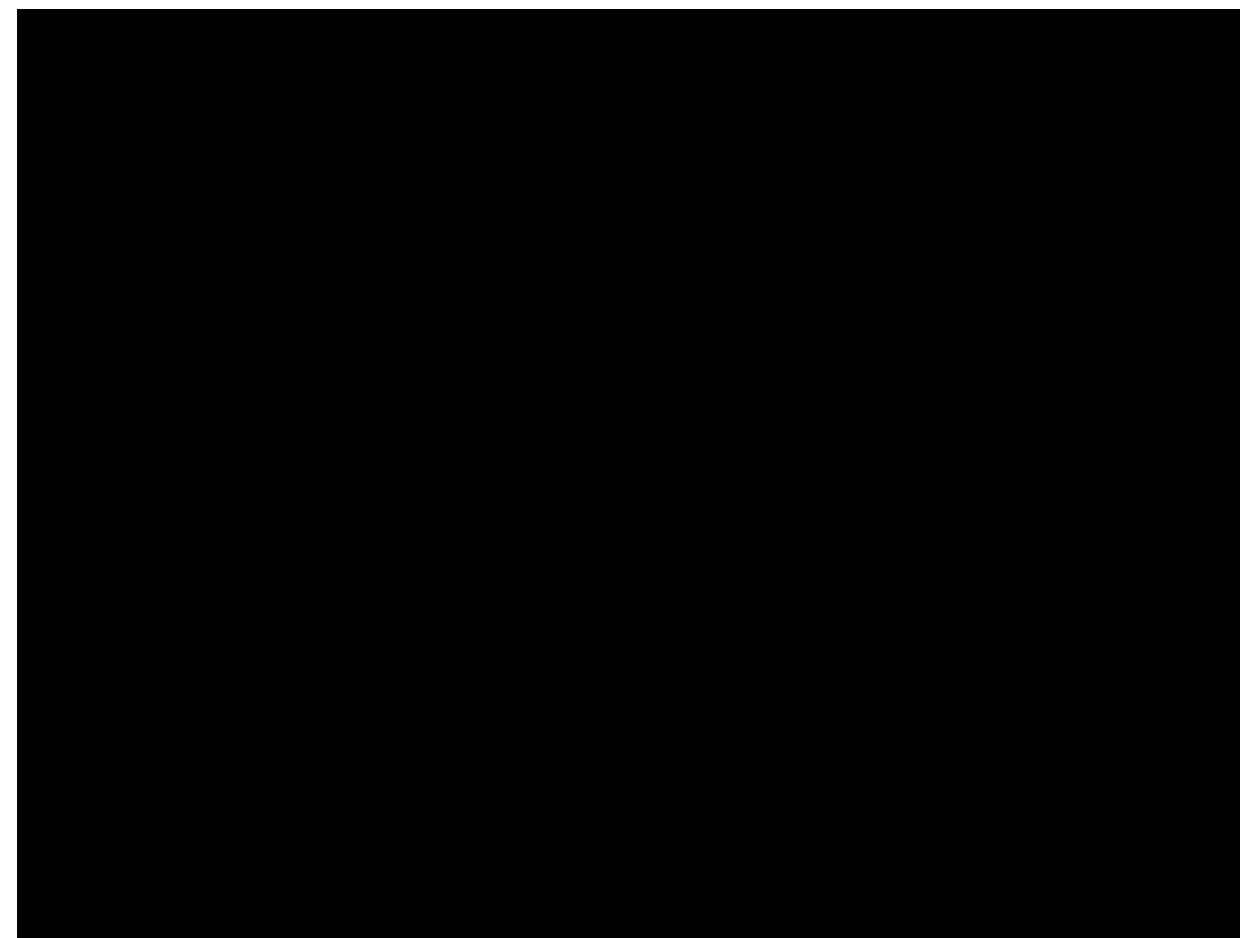


Local missing grid.

Original



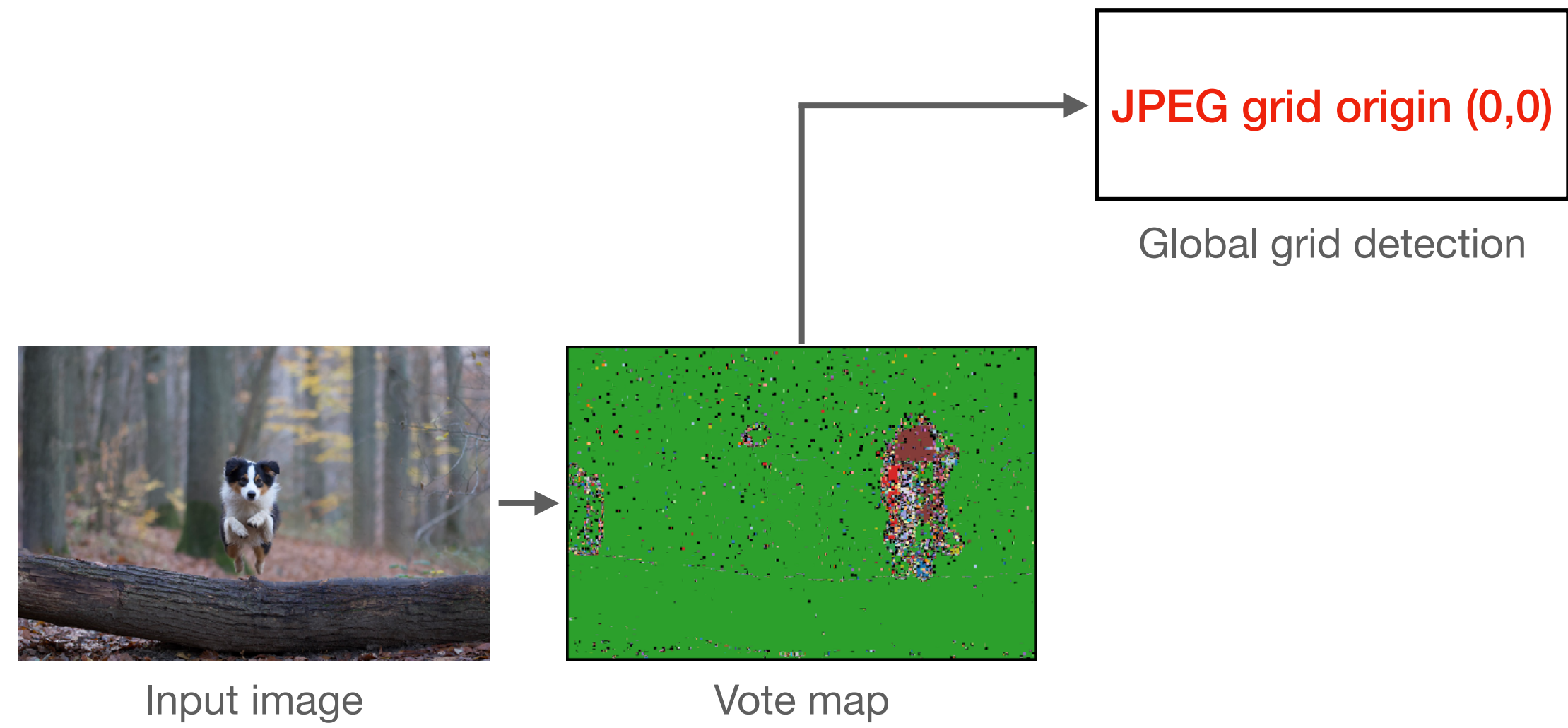
Forged



Detection results.



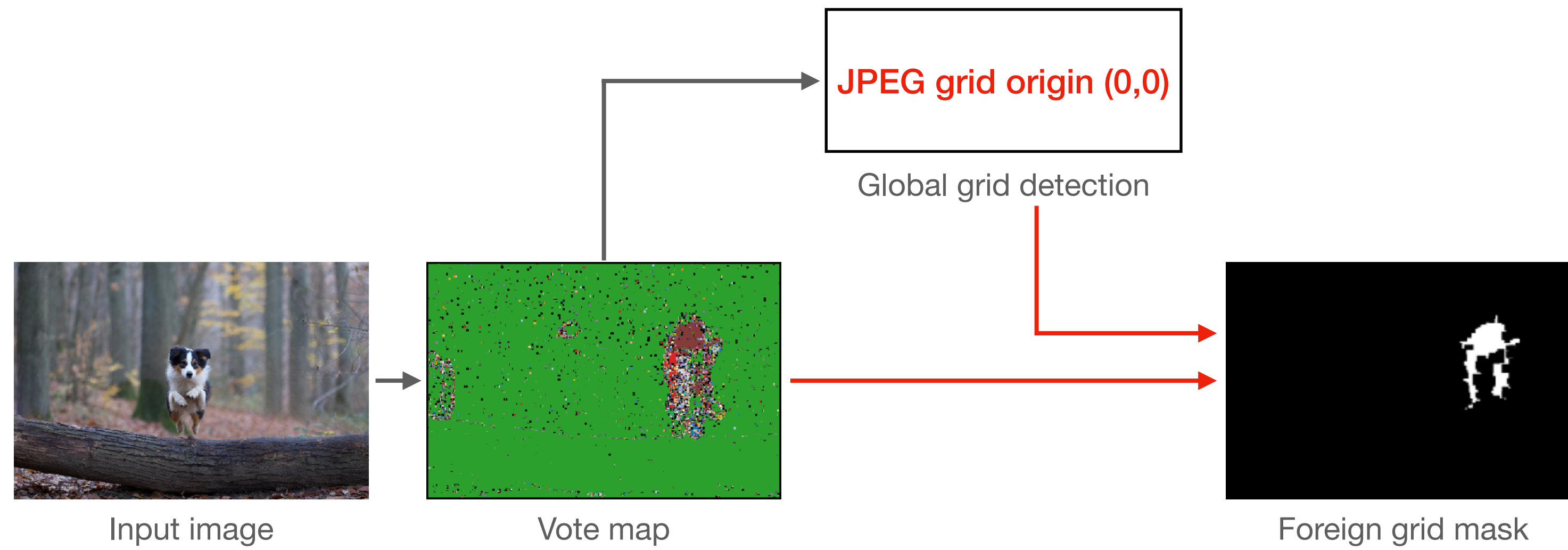
# ZERO



**JPEG grid detection based on the number of DCT zeros and its application to automatic and localized forgery detection.** T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.-M. Morel, and R. Grompone von Gioi. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRw), 2019.

**ZERO: a Local JPEG Grid Origin Detector Based on the Number of DCT Zeros and its Applications in Image Forensics.** T. Nikoukhah, J. Anger, M. Colom, J.-M. Morel, and R. Grompone von Gioi. Image Processing On Line, 2021.

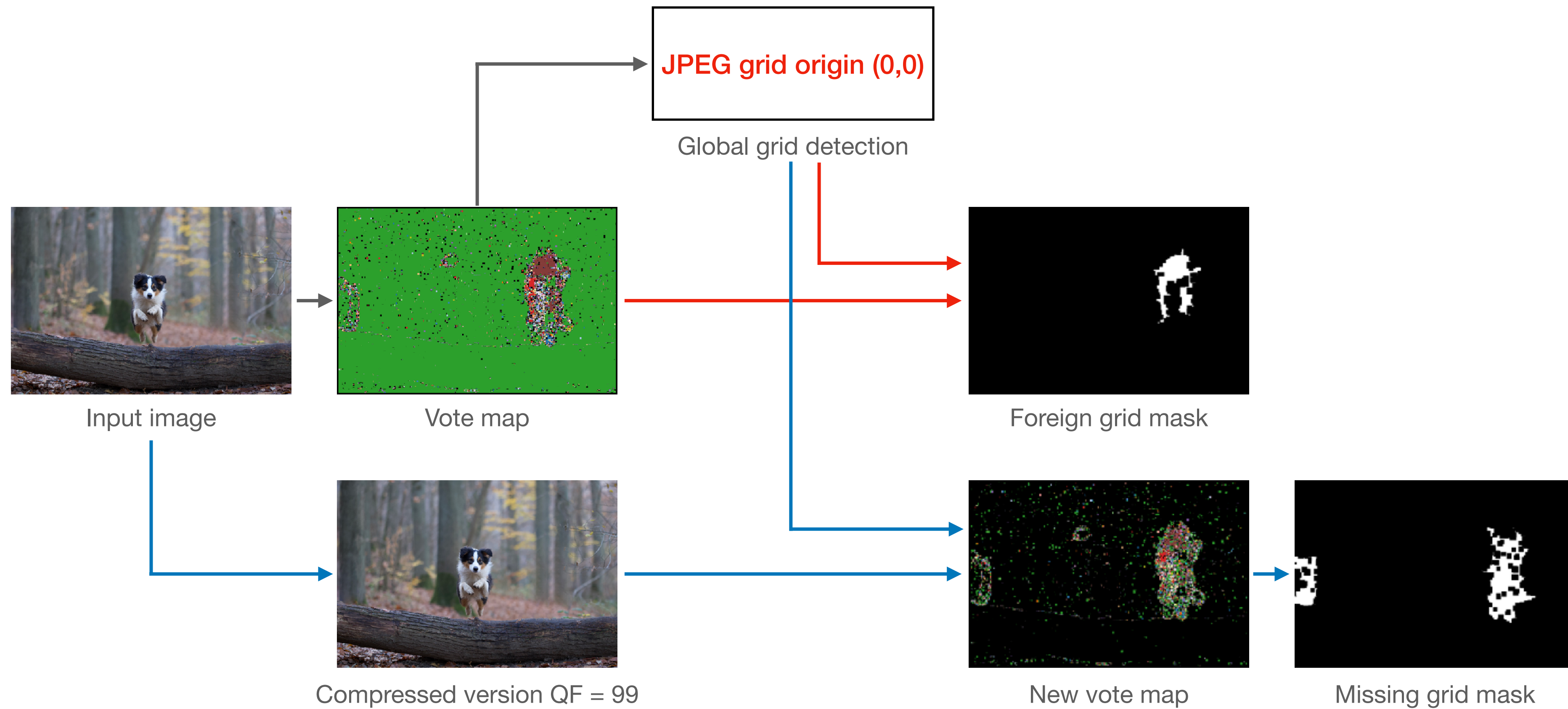
# ZERO



**JPEG grid detection based on the number of DCT zeros and its application to automatic and localized forgery detection.** T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.-M. Morel, and R. Grompone von Gioi. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRw), 2019.

**ZERO: a Local JPEG Grid Origin Detector Based on the Number of DCT Zeros and its Applications in Image Forensics.** T. Nikoukhah, J. Anger, M. Colom, J.-M. Morel, and R. Grompone von Gioi. Image Processing On Line, 2021.

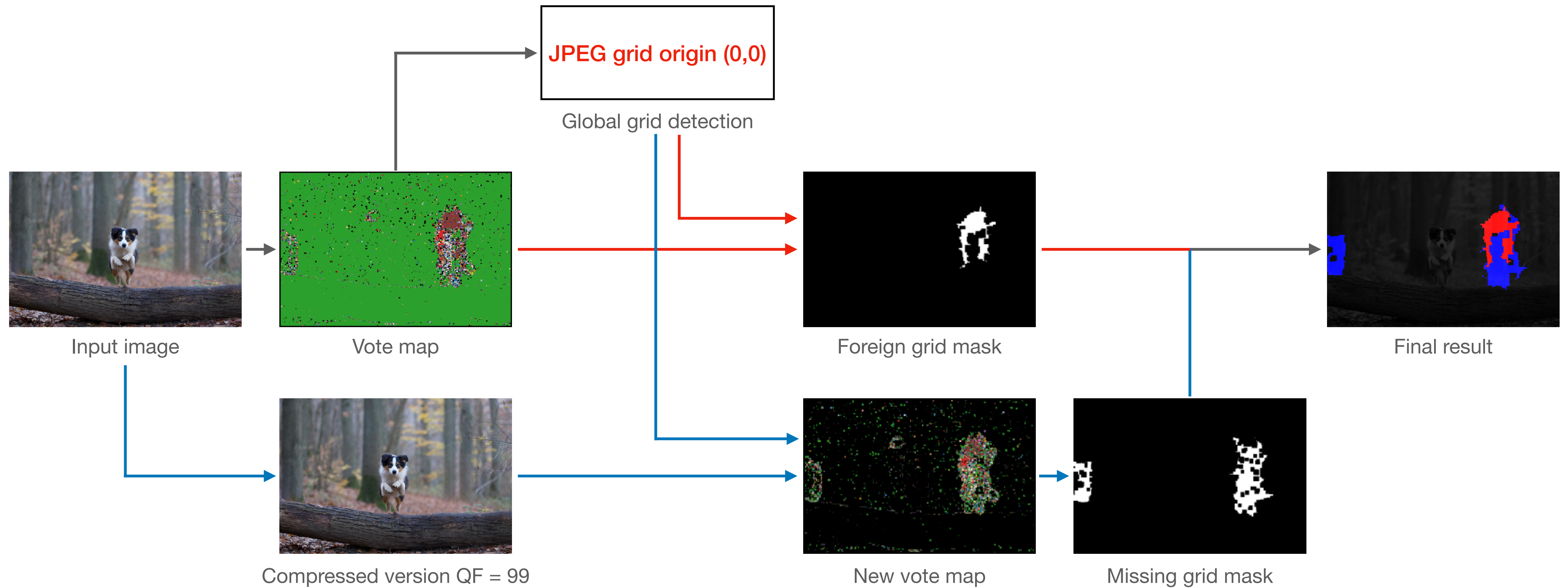
# ZERO



**JPEG grid detection based on the number of DCT zeros and its application to automatic and localized forgery detection.** T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.-M. Morel, and R. Grompone von Gioi. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRw), 2019.

**ZERO: a Local JPEG Grid Origin Detector Based on the Number of DCT Zeros and its Applications in Image Forensics.** T. Nikoukhah, J. Anger, M. Colom, J.-M. Morel, and R. Grompone von Gioi. Image Processing On Line, 2021.

# ZERO

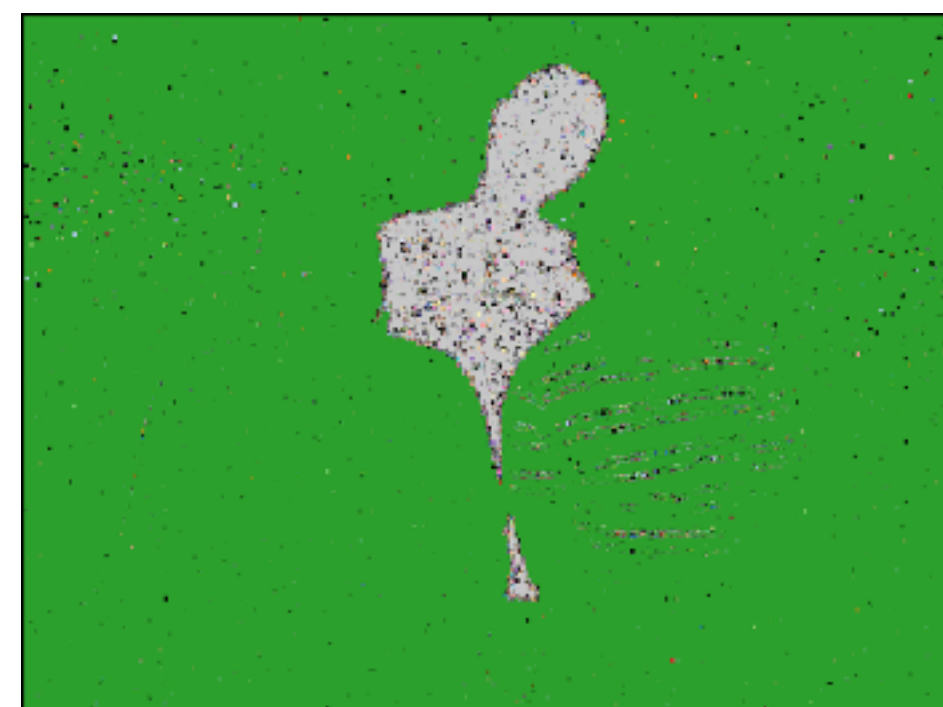
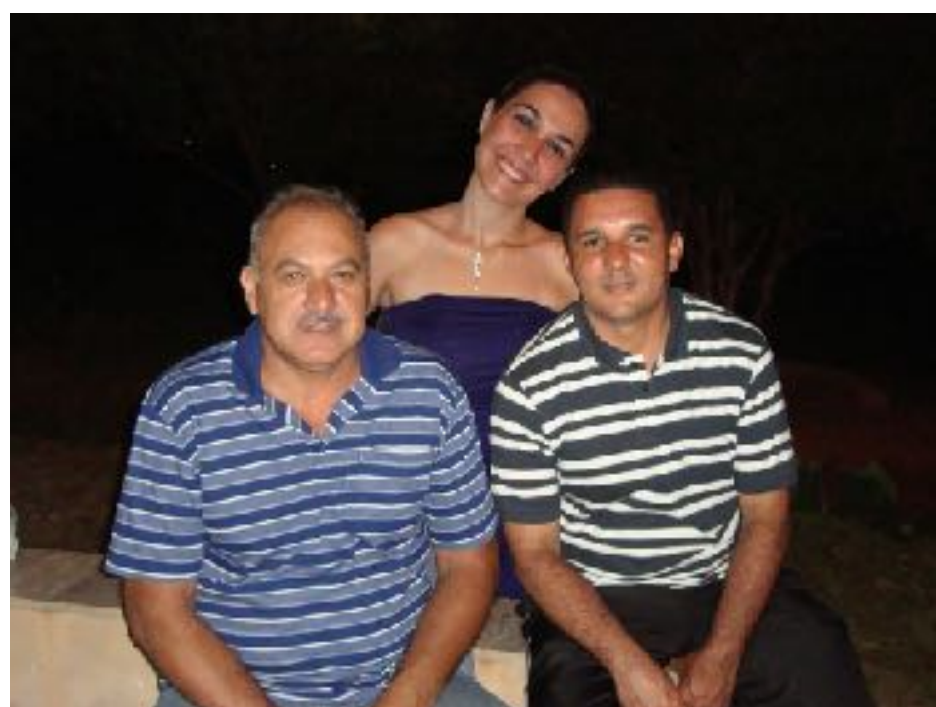


**JPEG grid detection based on the number of DCT zeros and its application to automatic and localized forgery detection.** T. Nikoukhah, J. Anger, T. Ehret, M. Colom, J.-M. Morel, and R. Grompone von Gioi. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRw), 2019.

**ZERO: a Local JPEG Grid Origin Detector Based on the Number of DCT Zeros and its Applications in Image Forensics.** T. Nikoukhah, J. Anger, M. Colom, J.-M. Morel, and R. Grompone von Gioi. Image Processing On Line, 2021.

# ZERO: a forgery detector

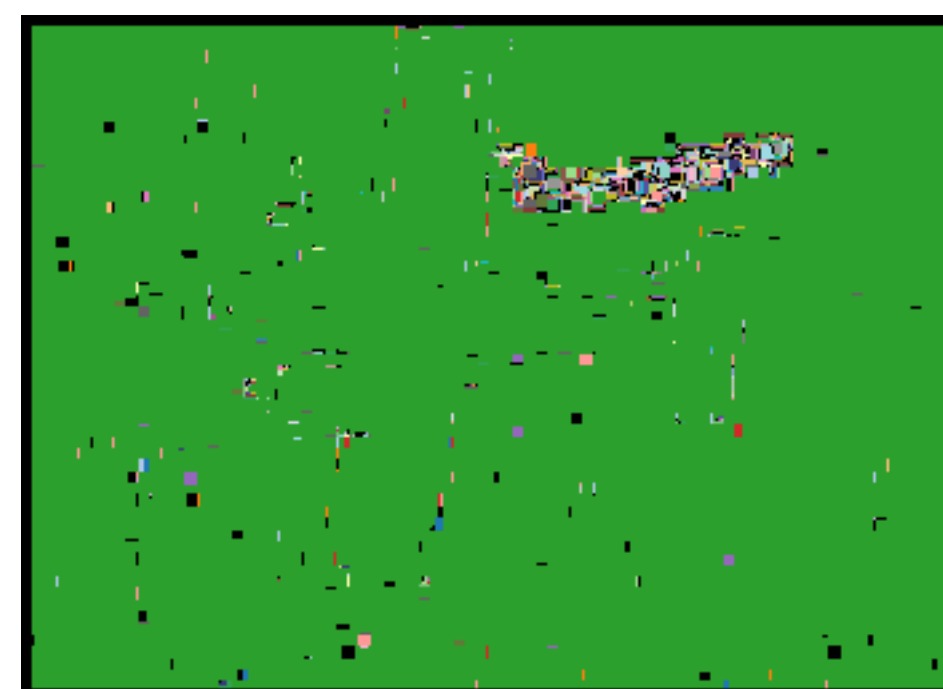
DSO-1 dataset



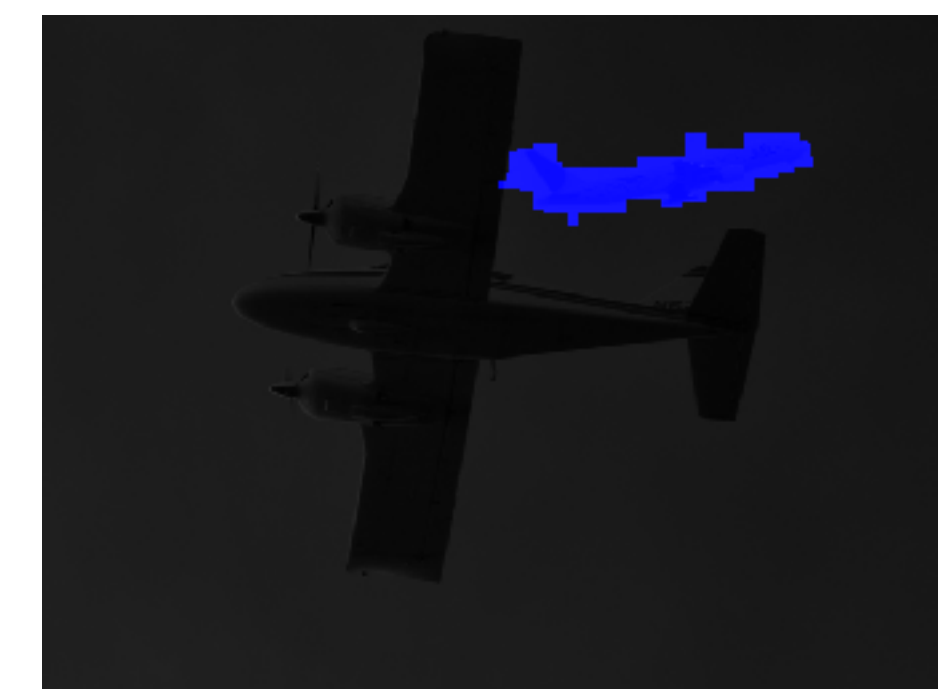
JPEG grid origin (0,0)



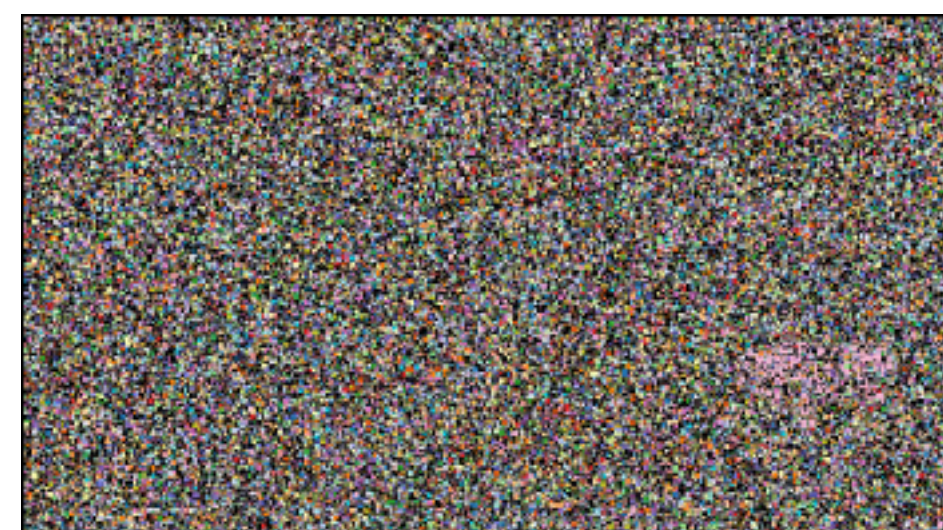
DEFACTO dataset



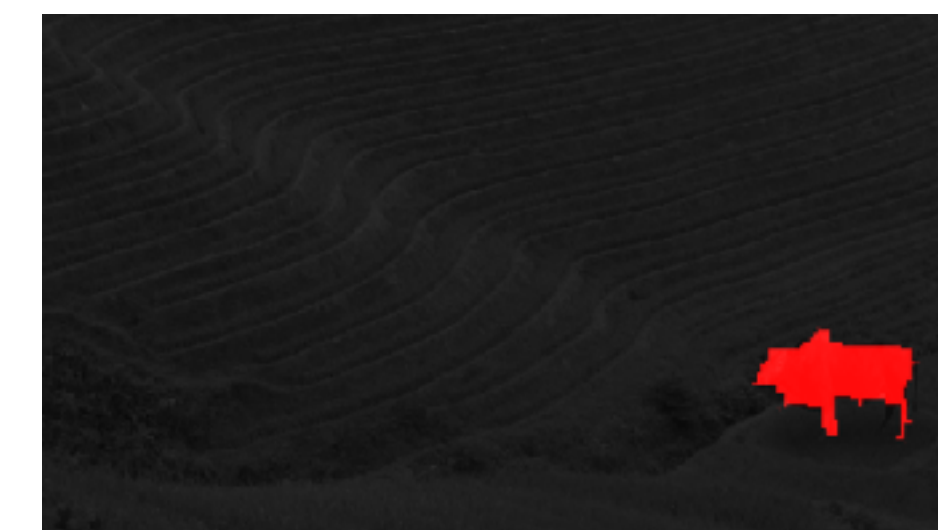
JPEG grid origin (0,0)



Korus dataset



No global JPEG grid

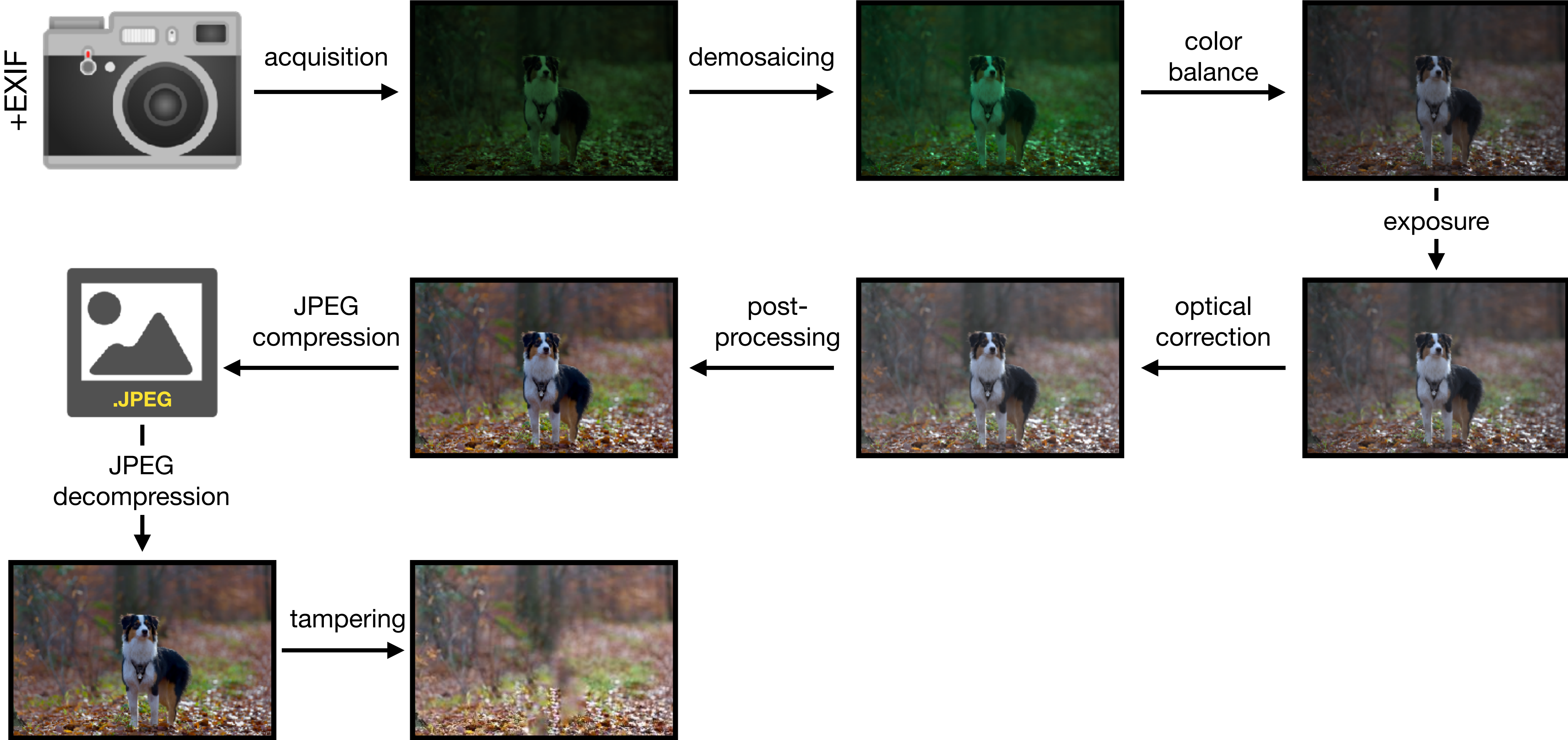


Forged images from datasets.

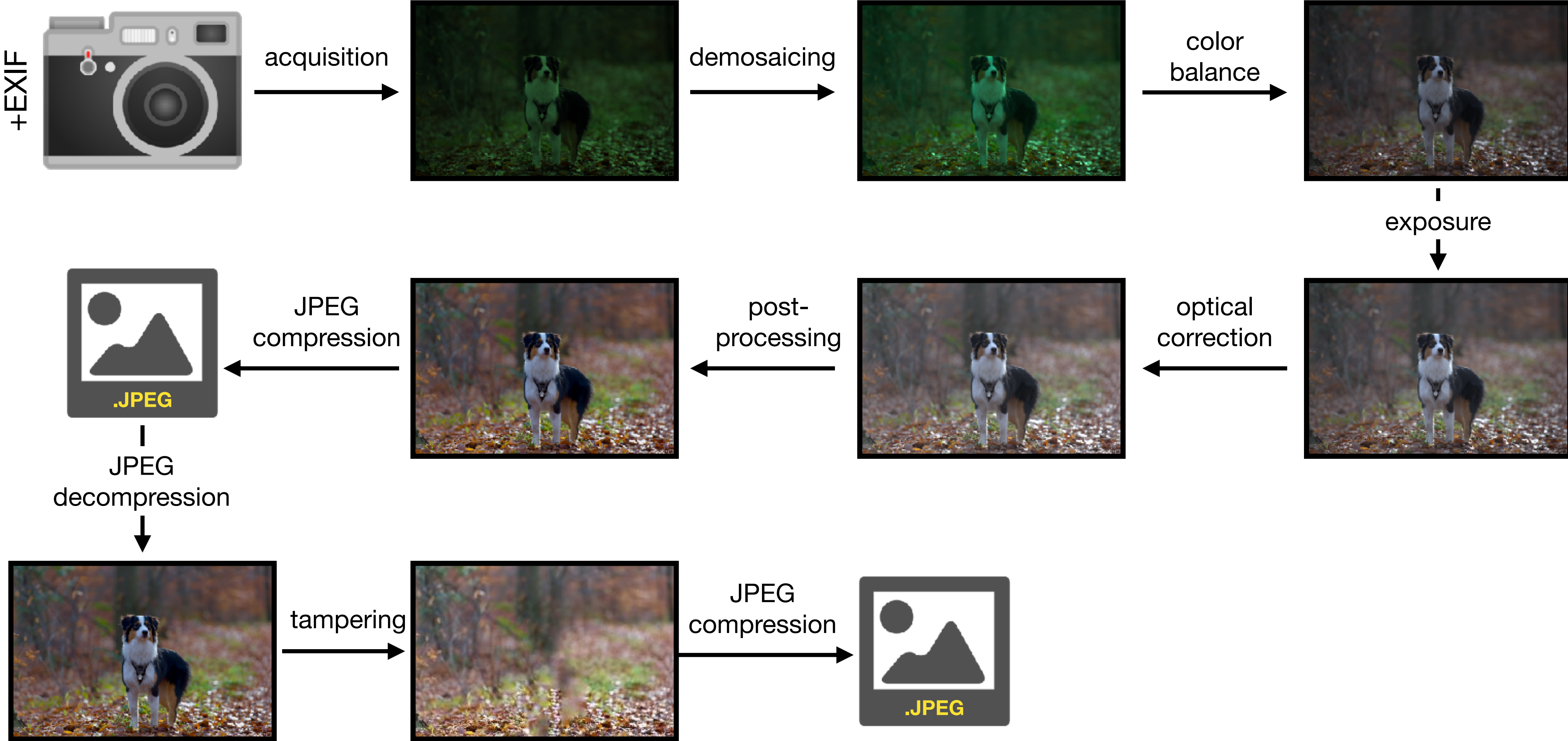
Vote maps

Detection results

# Image processing pipeline



# Image processing pipeline



# Forged images posted online

Original



Inpainting



Copy-move

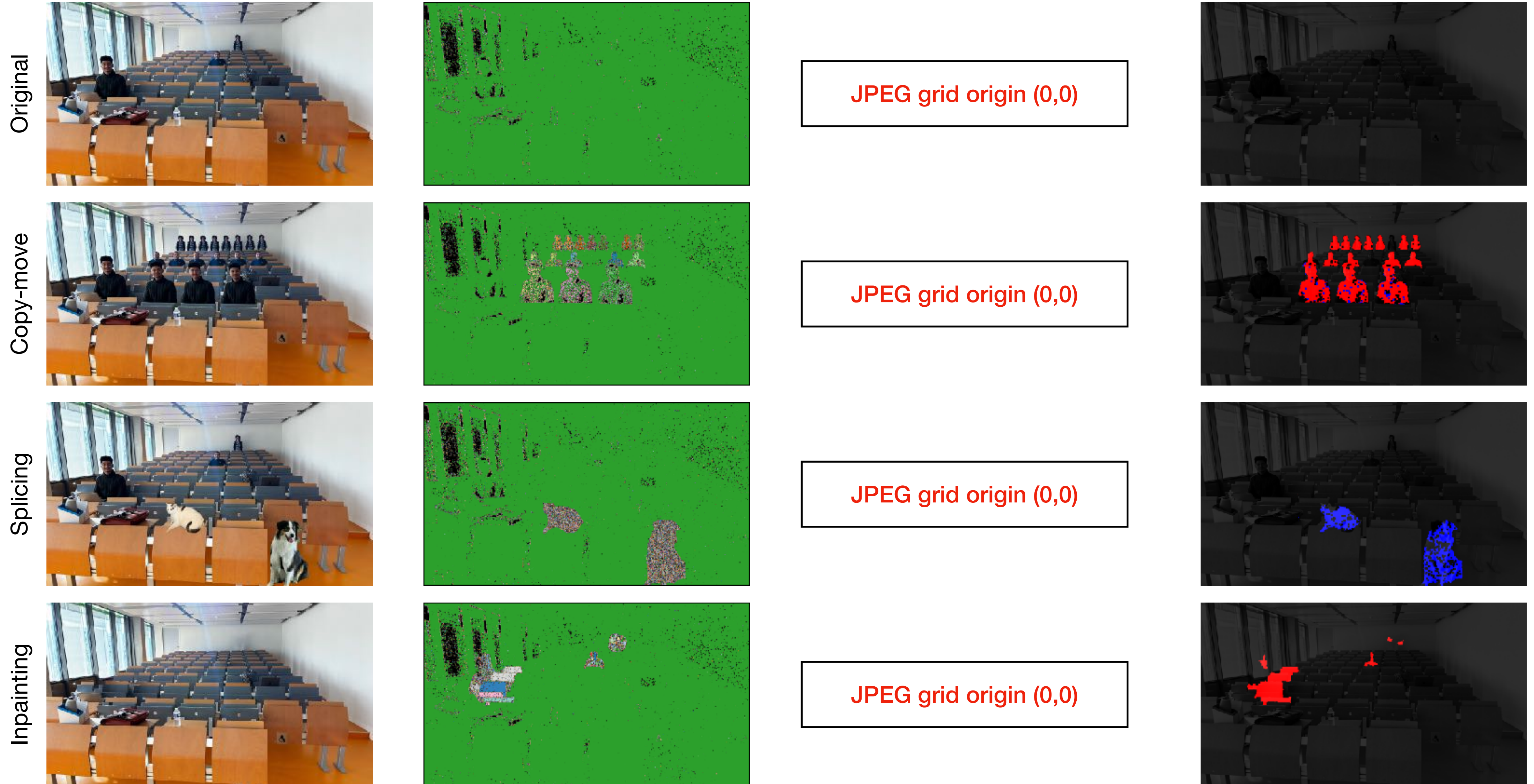


Splicing

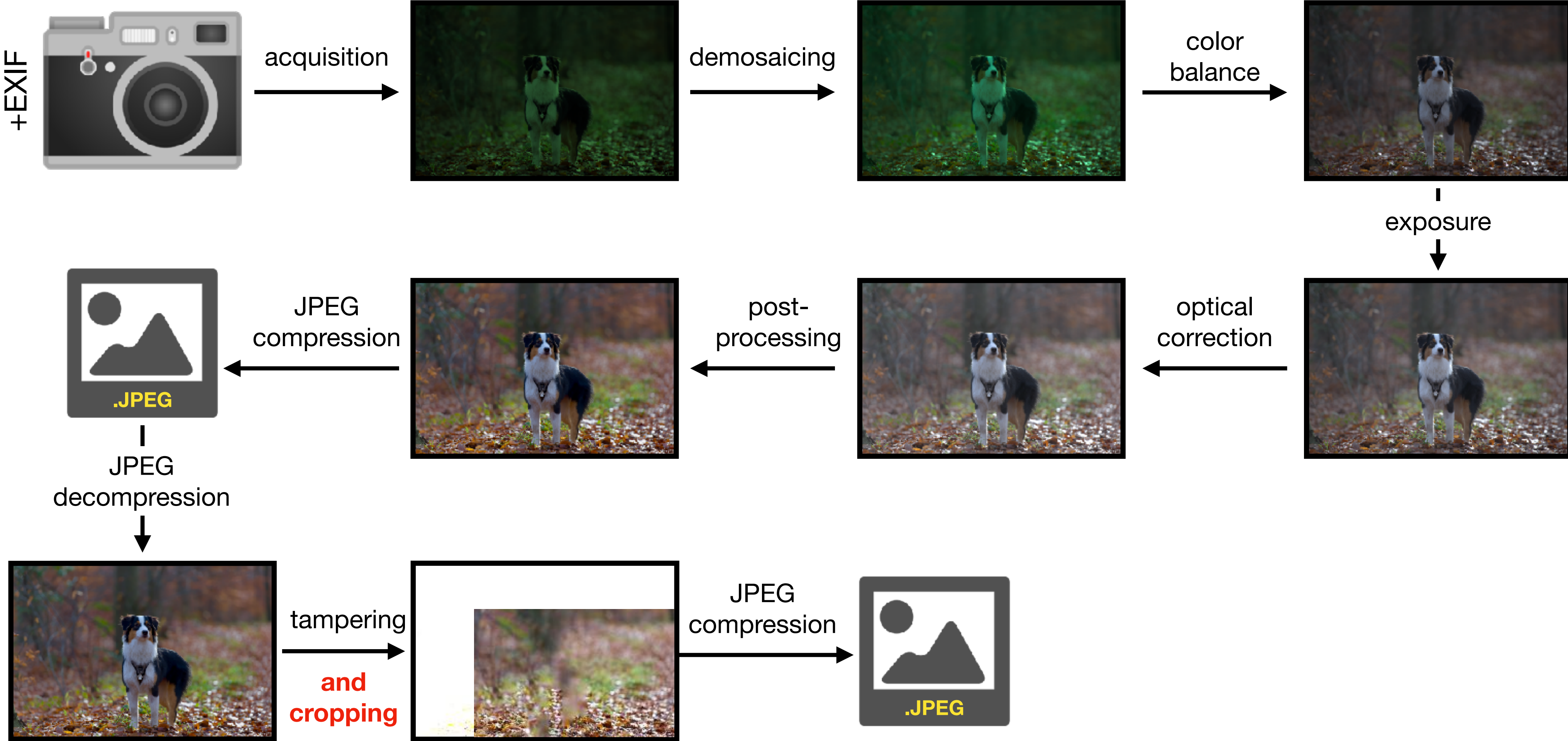




# ZERO: a forgery detector



# Image processing pipeline



# Forged image posted online



Tina   
@TinaNkh

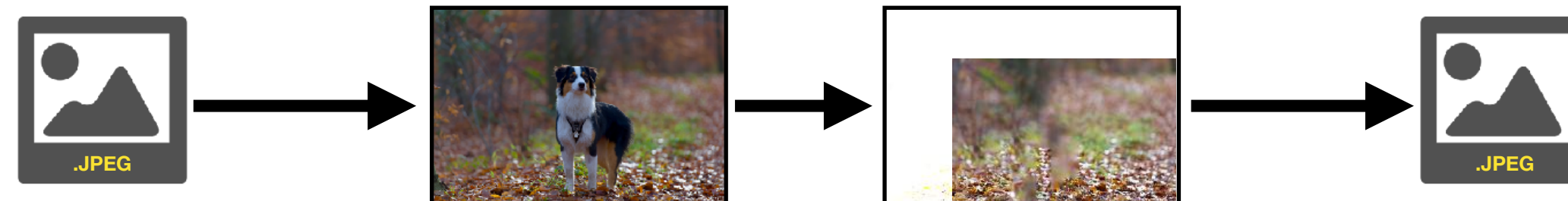


Can you detect these forgeries? [#TinasPhdDefense](#)



4:10 PM · Nov 8, 2022 · Twitter Web App

# Forged image

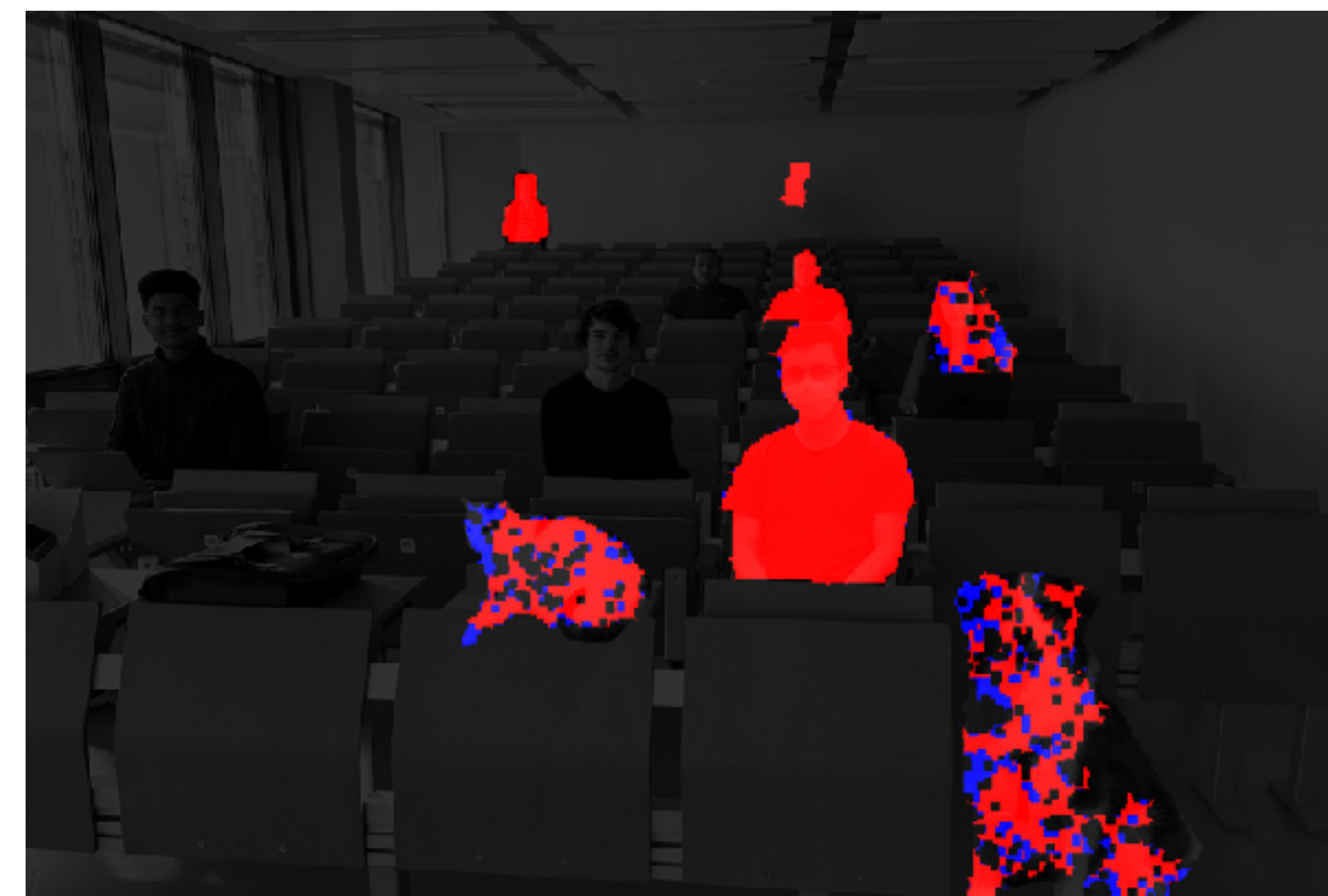


Forged image

JPEG grid origin (2,5)  
JPEG grid origin (0,0)  
  
Main grid (2,5)



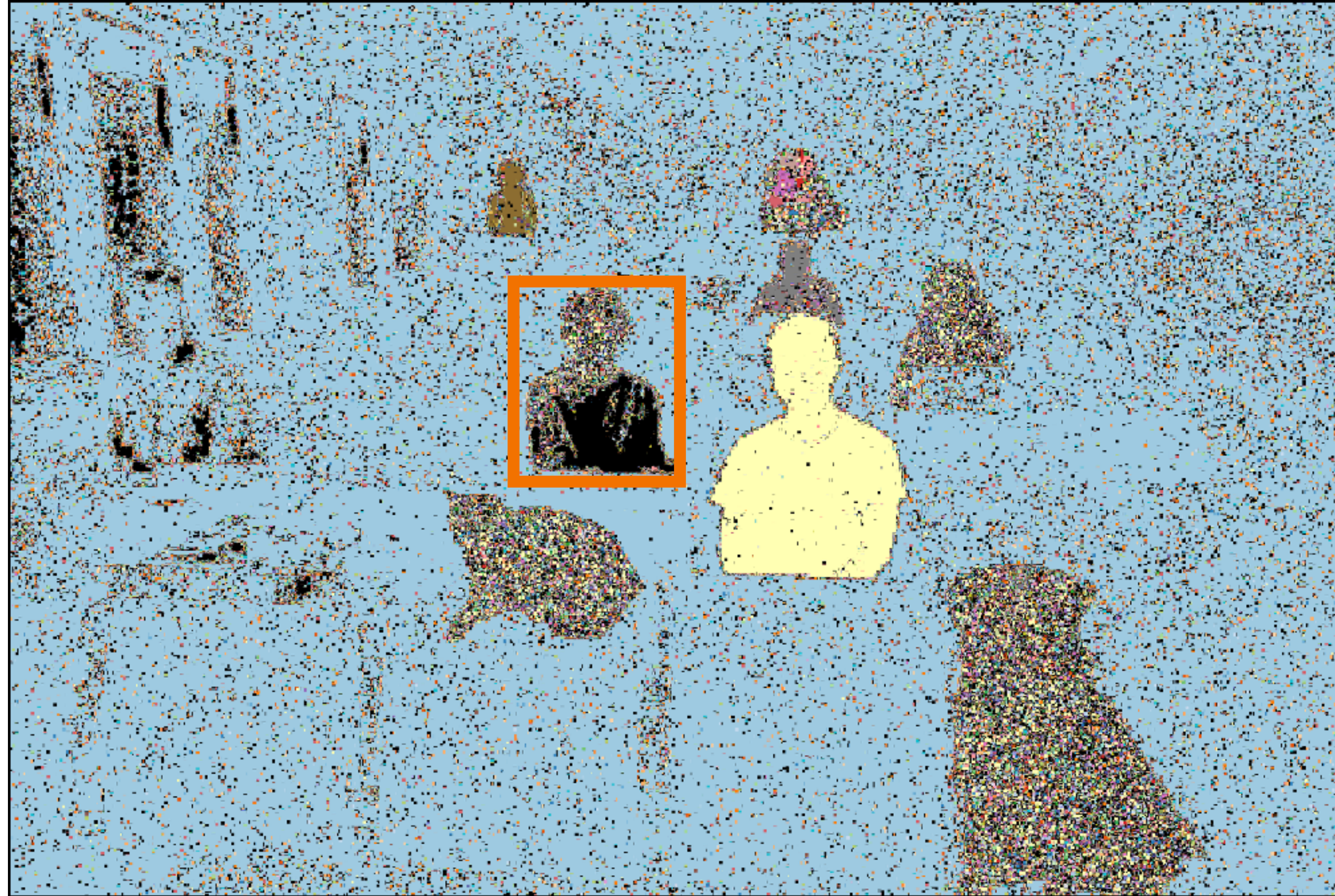
Vote map



Detection result

# ZERO: missed detections

Limitations: saturation, size



Vote map: each color represents a grid origin. Black corresponds to a non-valid vote, in case of a tie for example.

# Forged image posted online



Tina   
@TinaNkh



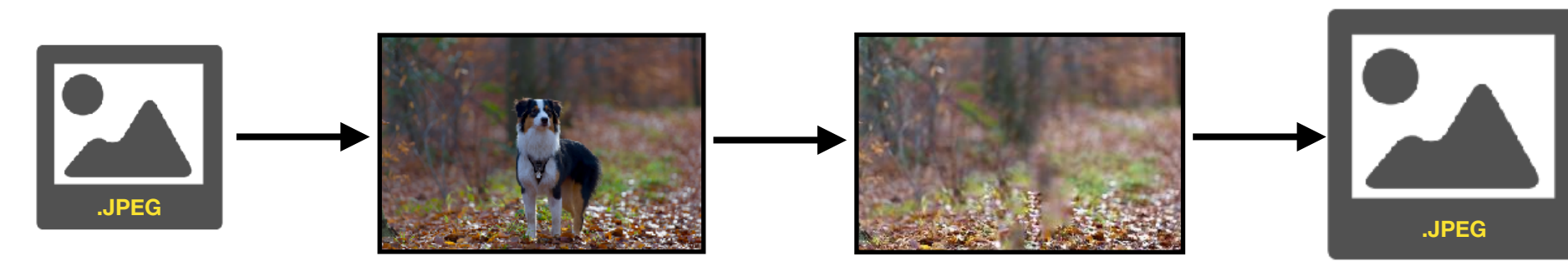
Can you detect these forgeries? [#TinasPhdDefense](#)



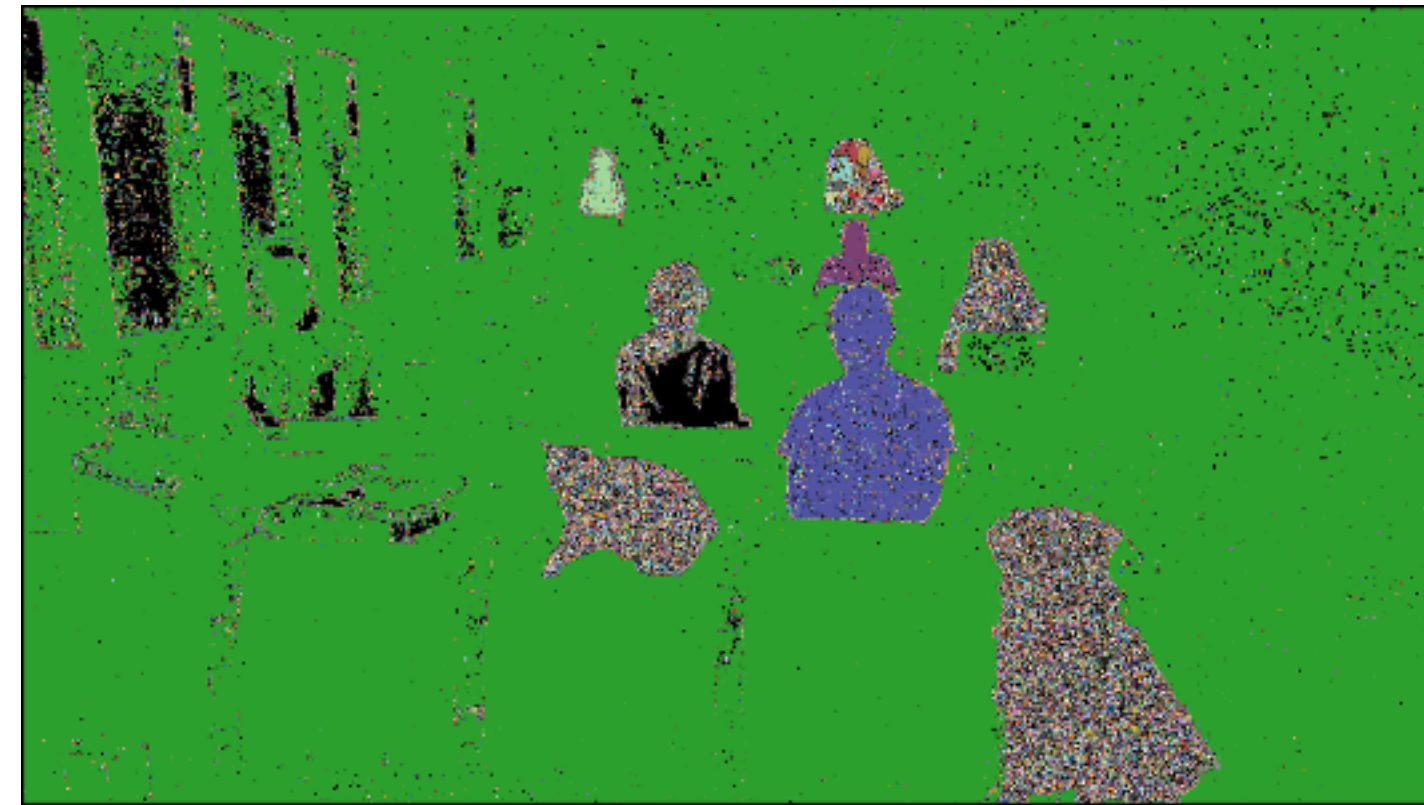
4:10 PM · Nov 8, 2022 · Twitter Web App

# ZERO and double compression

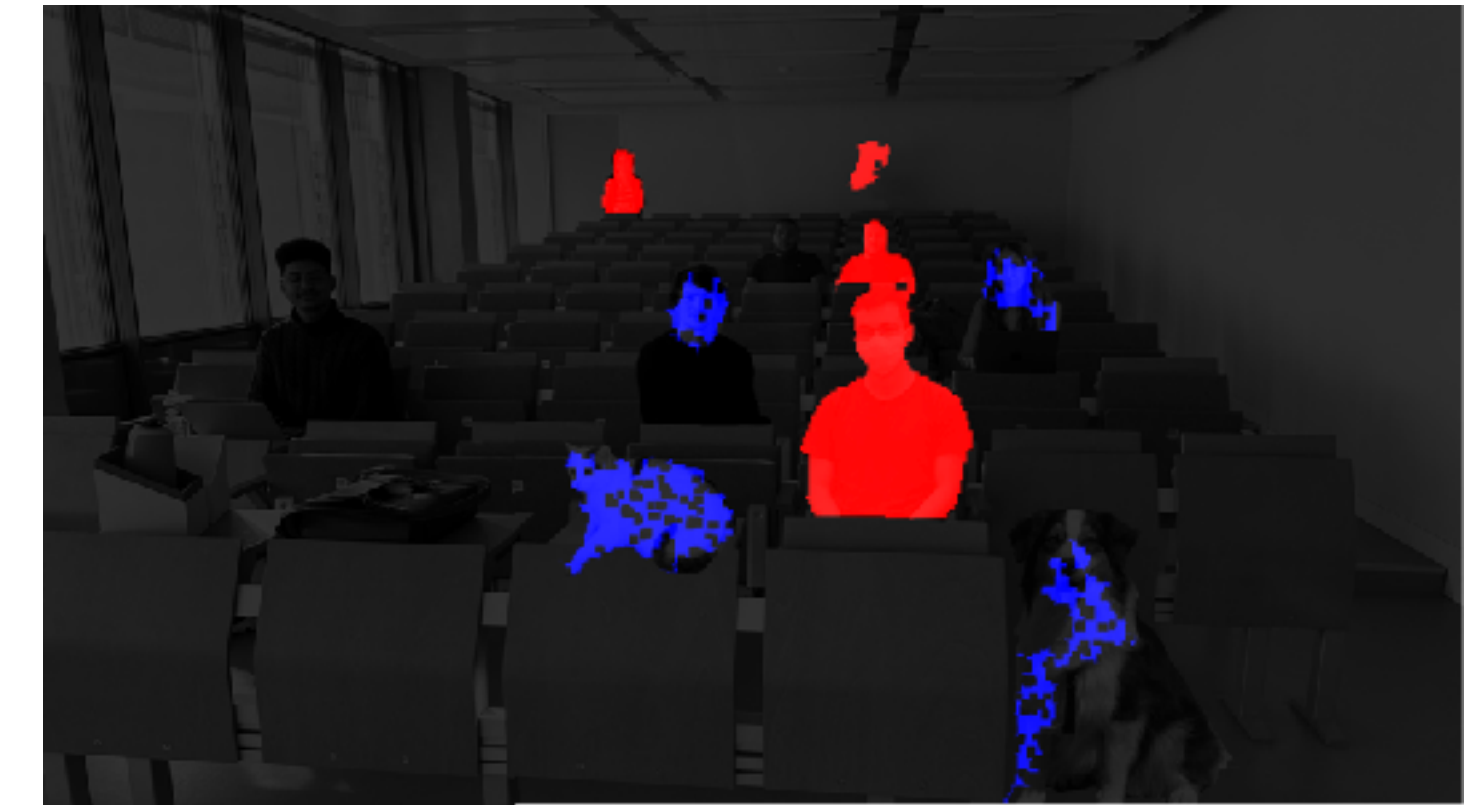
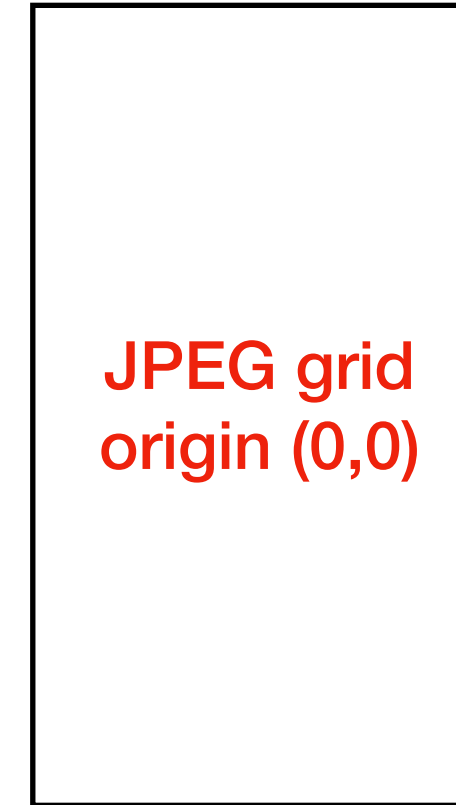
Second compression QF2 = 100



Forged image



Vote map



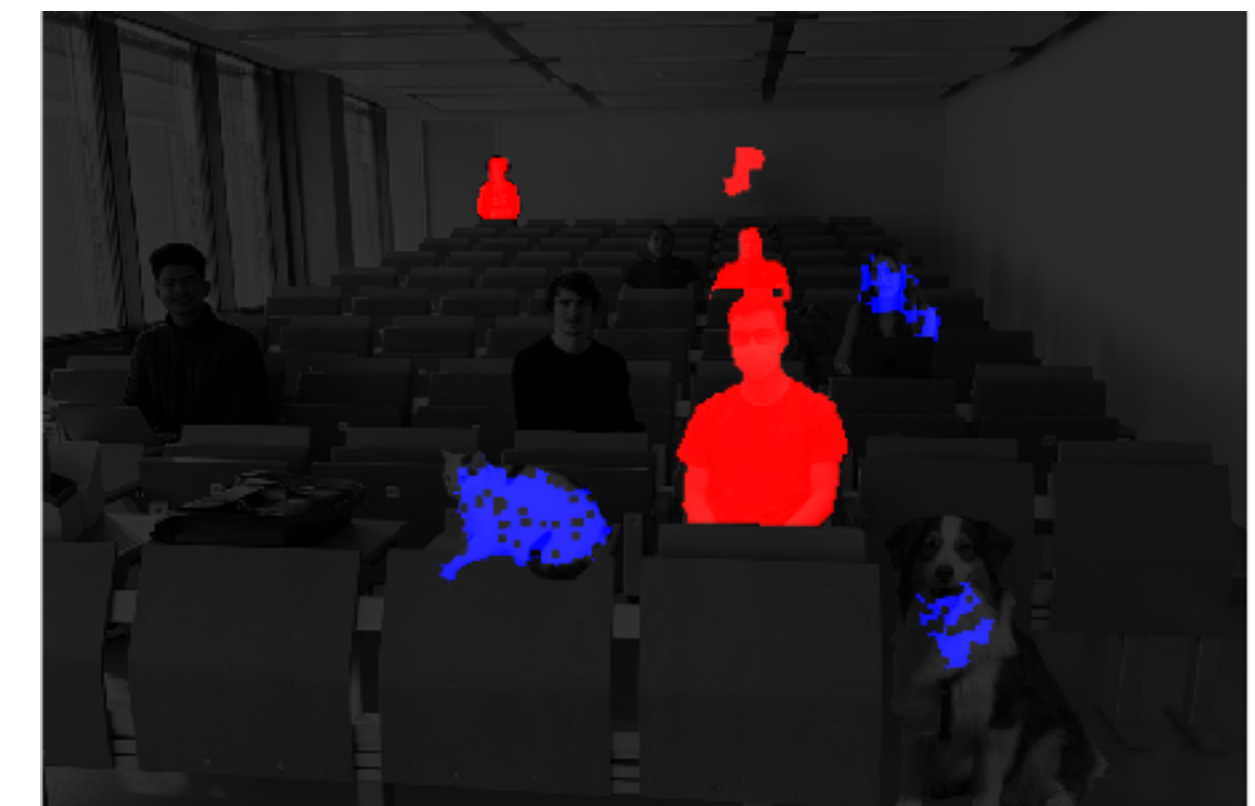
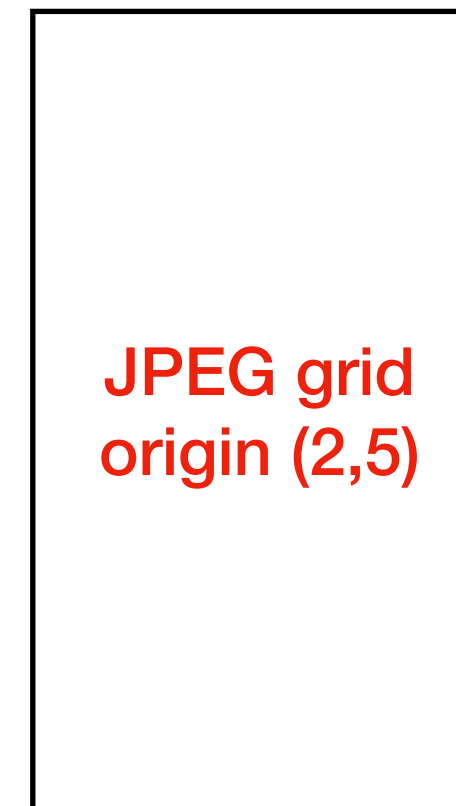
Detection result



Forged and cropped image



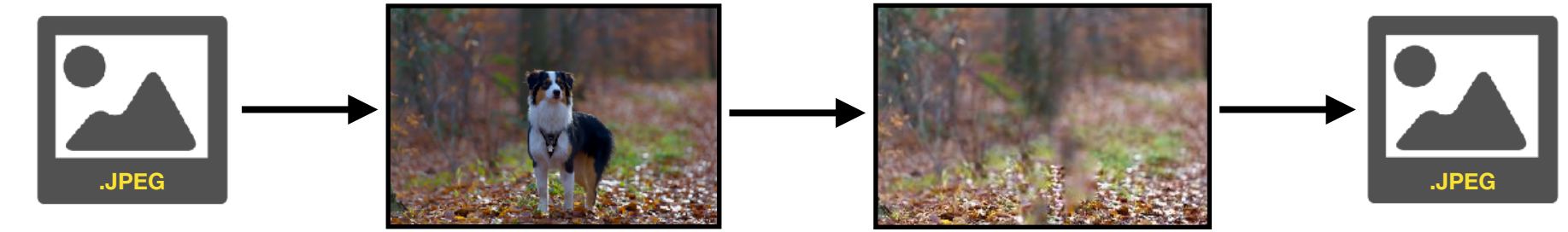
Vote map



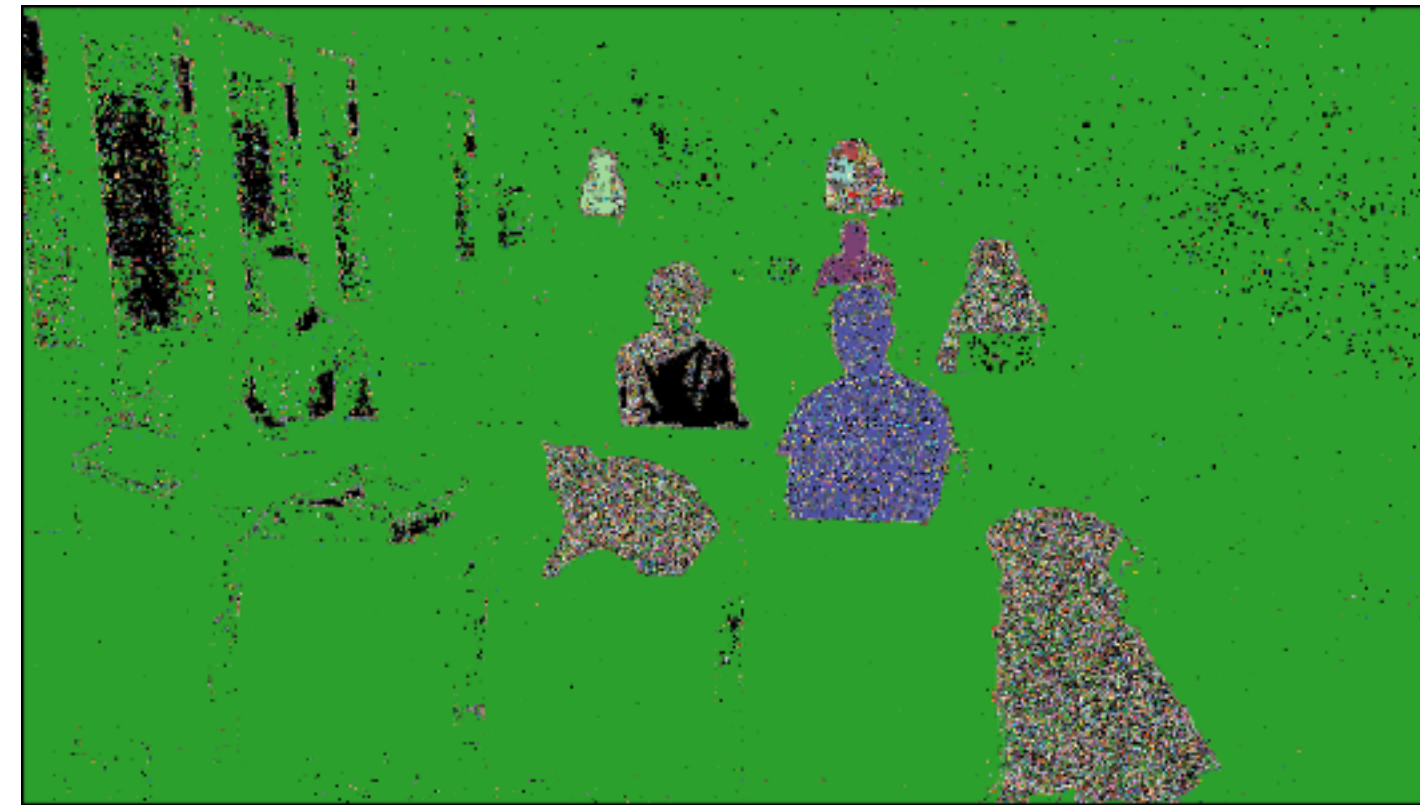
Detection result

# ZERO and double compression

Second compression  $QF2 \geq QF1$

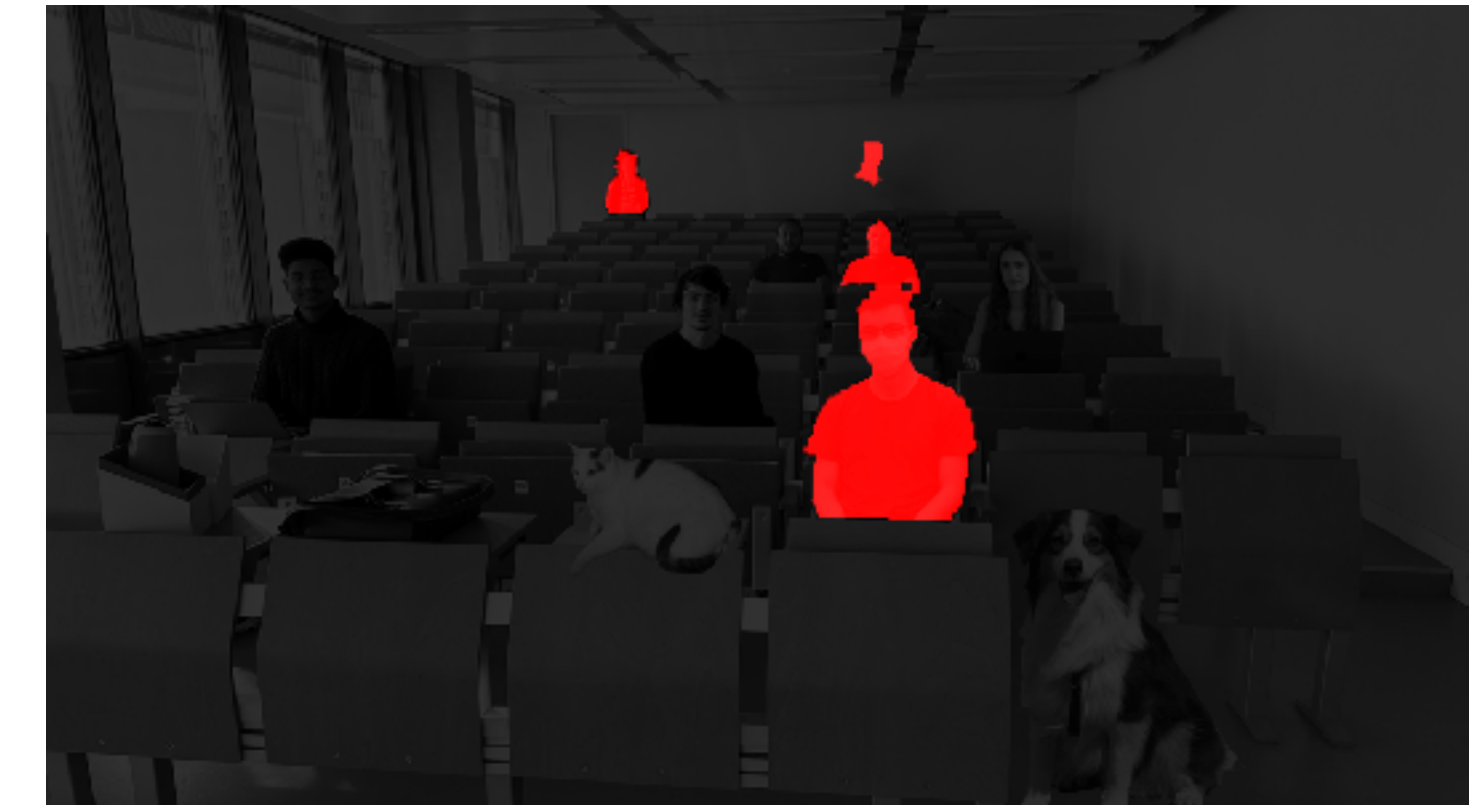


Forged image



Vote map

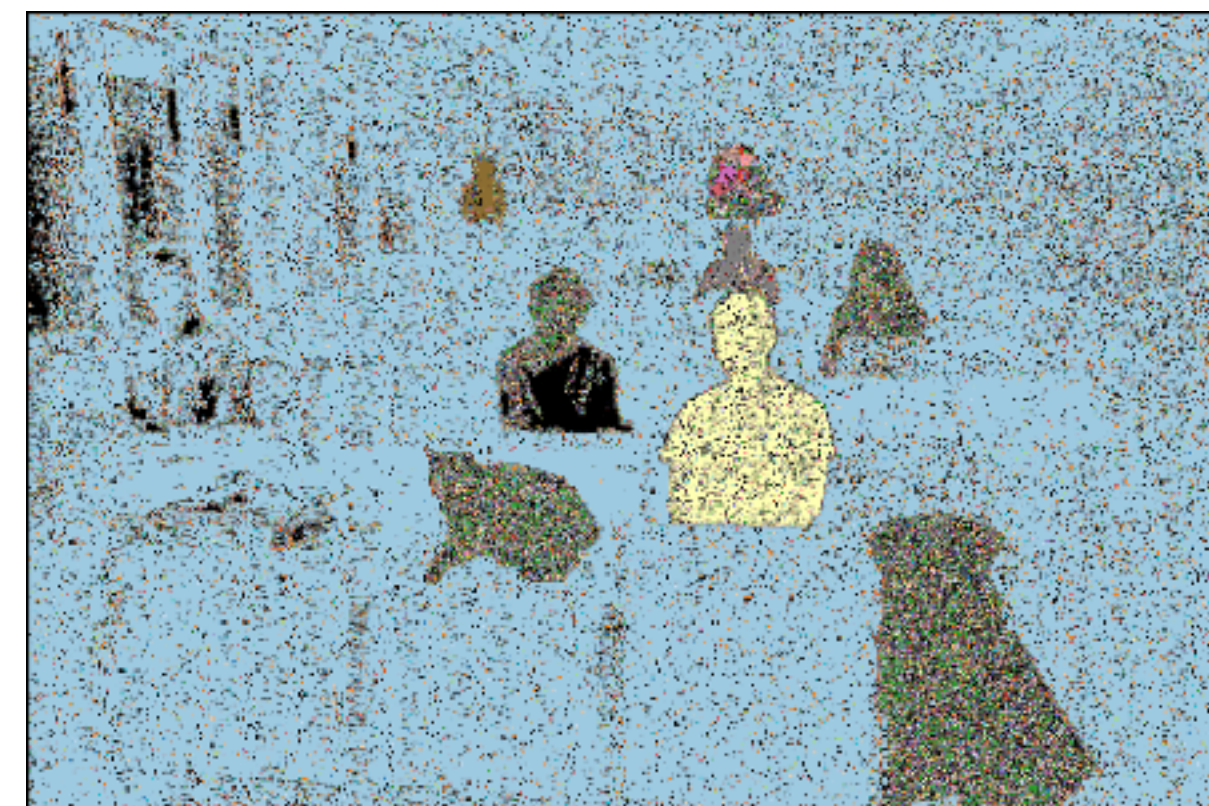
JPEG grid origin (0,0)



Detection result

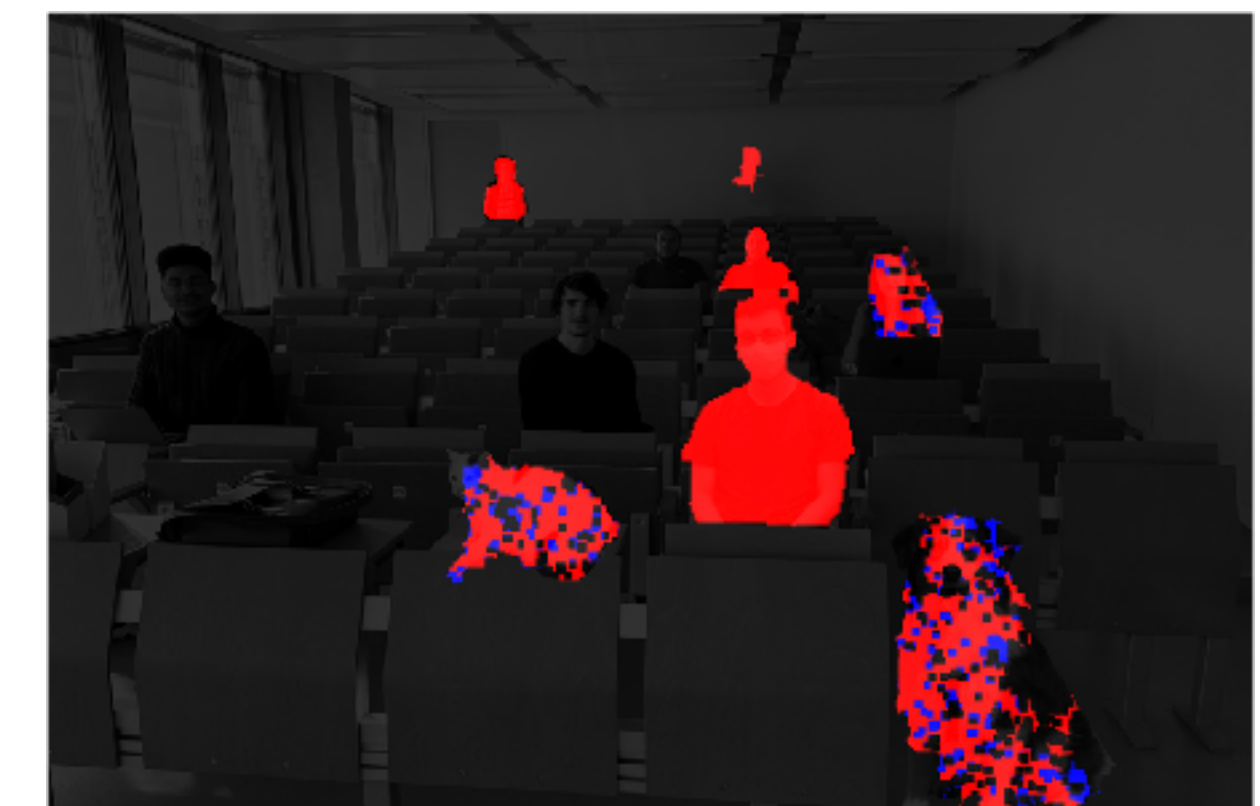


Forged and cropped image



Vote map

JPEG grid origin (2,5)  
JPEG grid origin (0,0)  
Main grid (2,5)

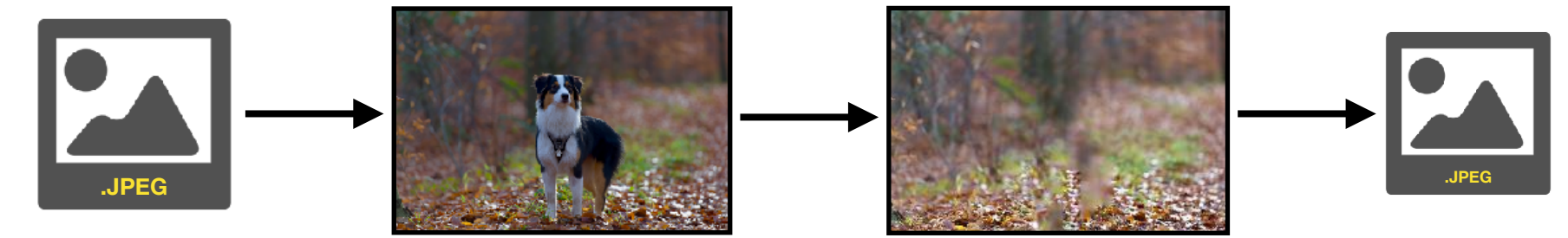


Detection result

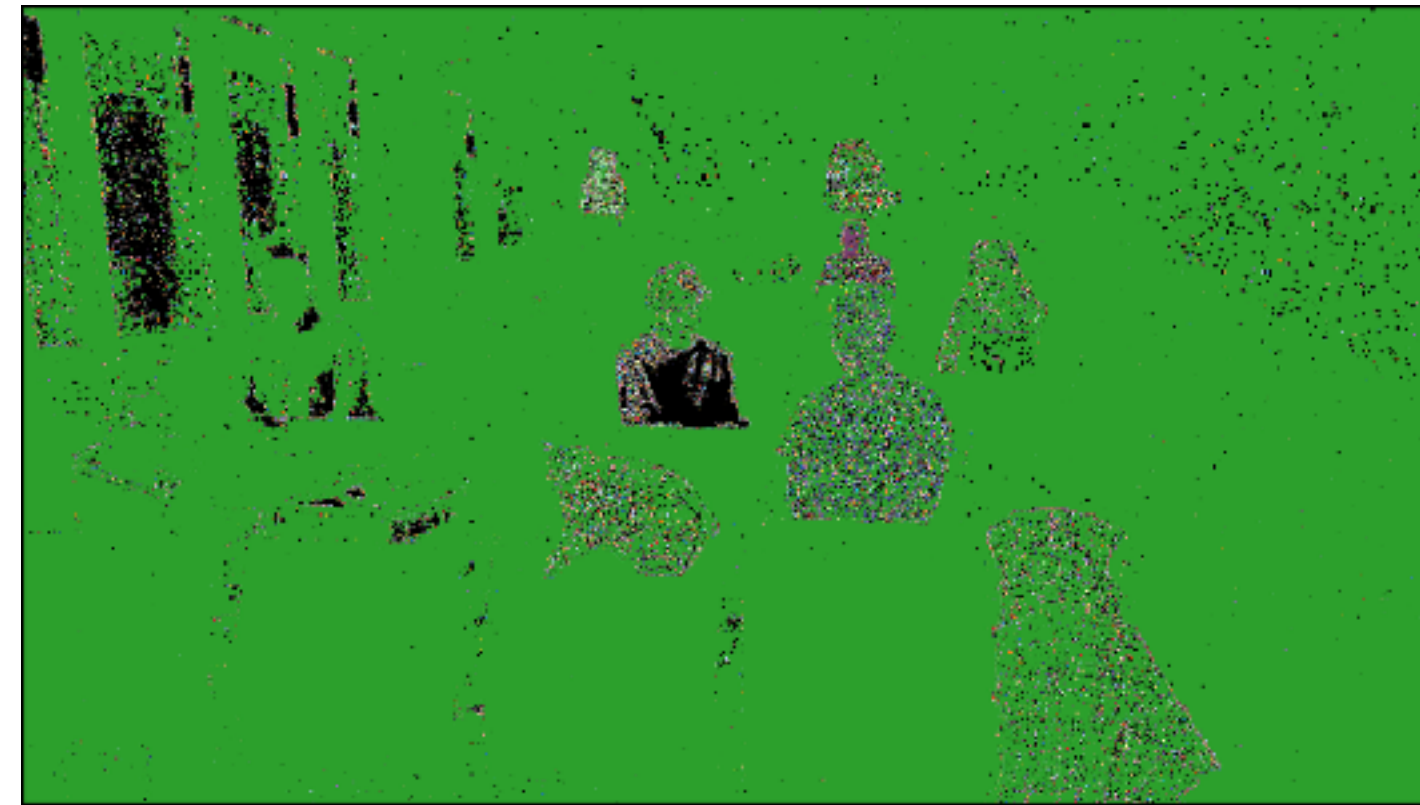


# ZERO and double compression

Second compression QF2 < QF1

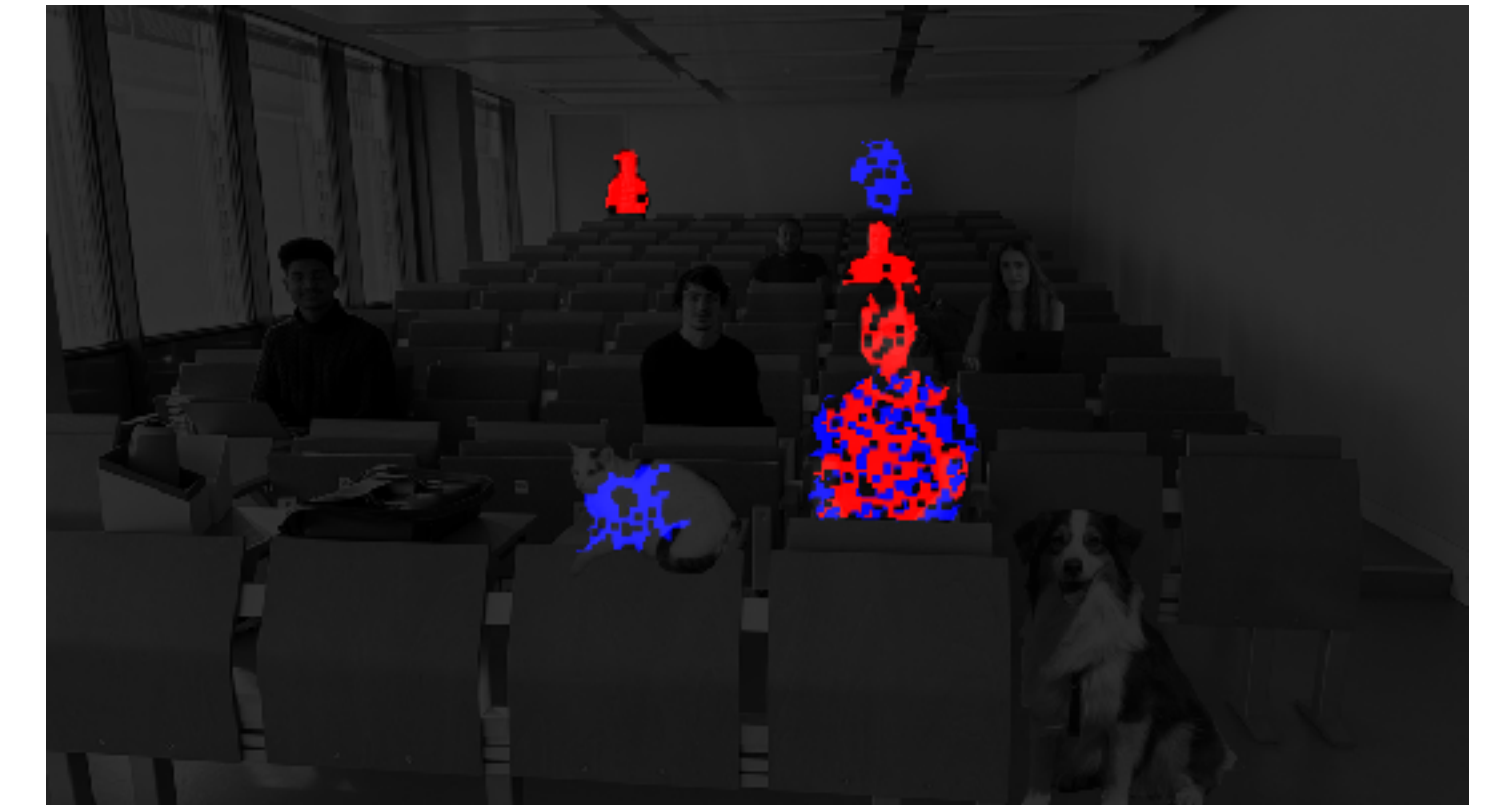


Forged image



Vote map

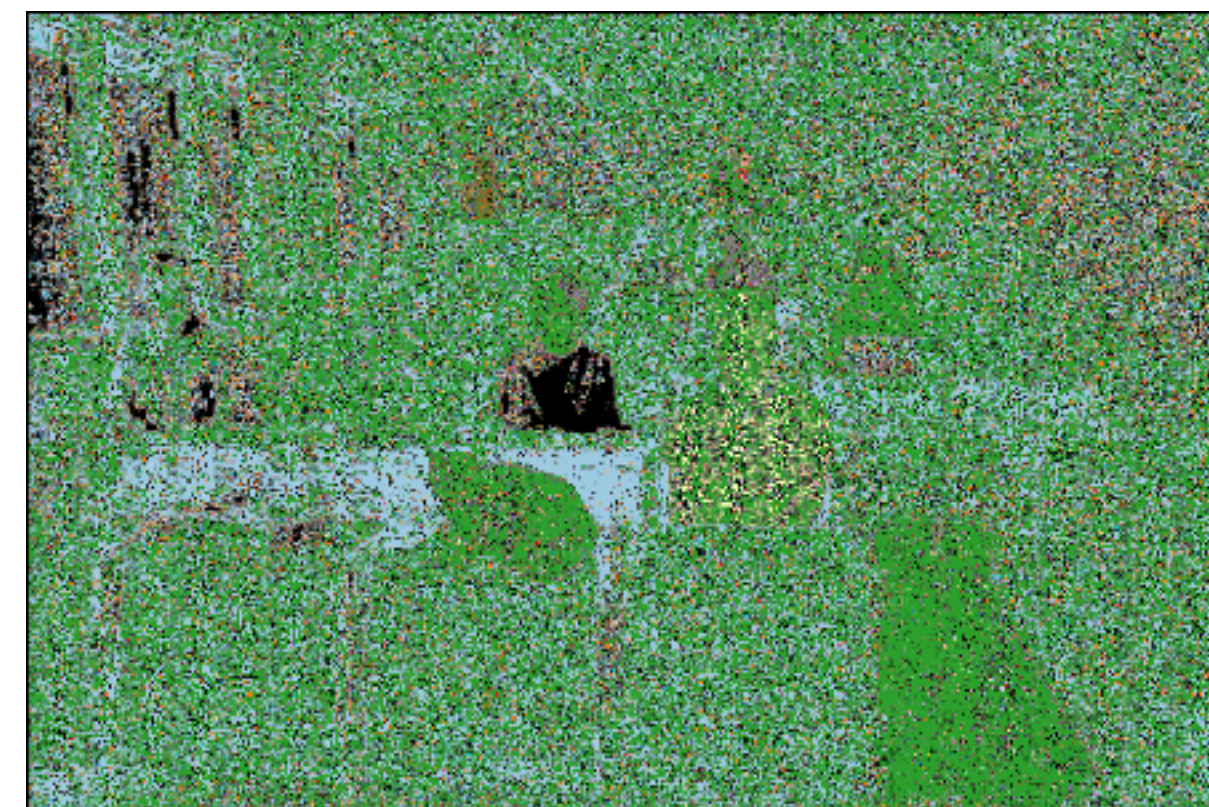
JPEG grid origin (0,0)



Detection result

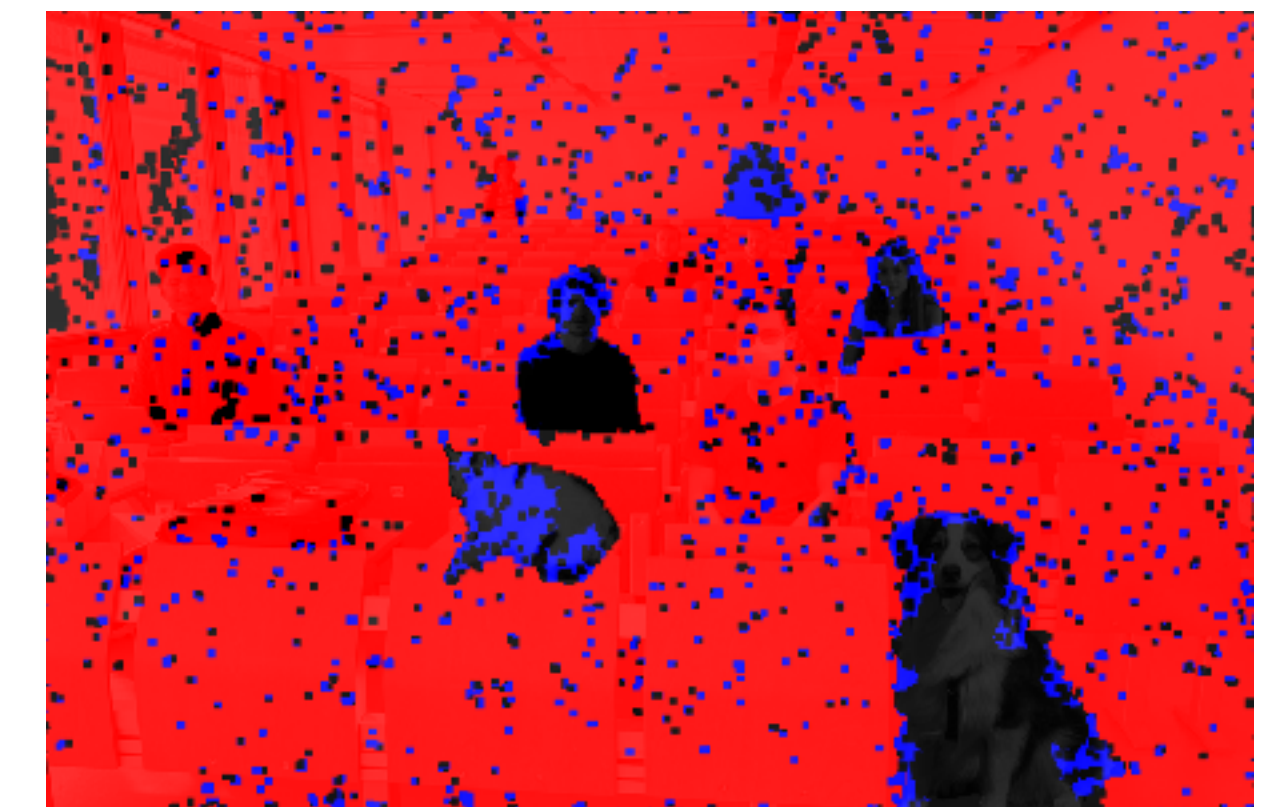


Forged and cropped image



Vote map

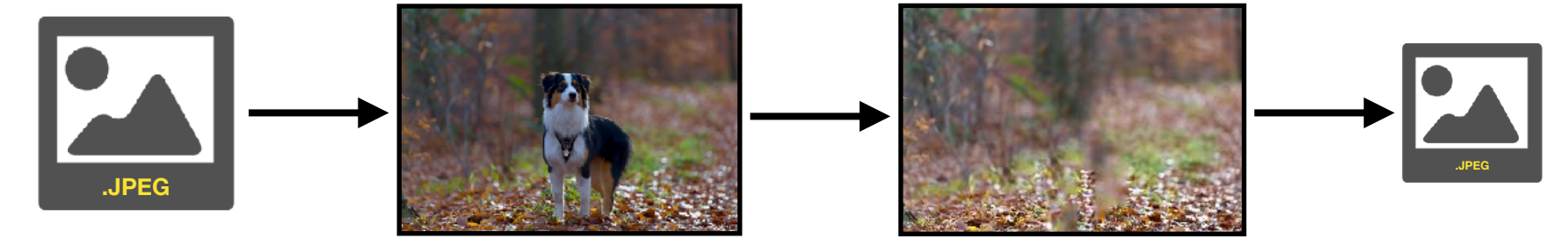
JPEG grid origin (0,0)  
JPEG grid origin (2,5)  
Main grid (0,0)



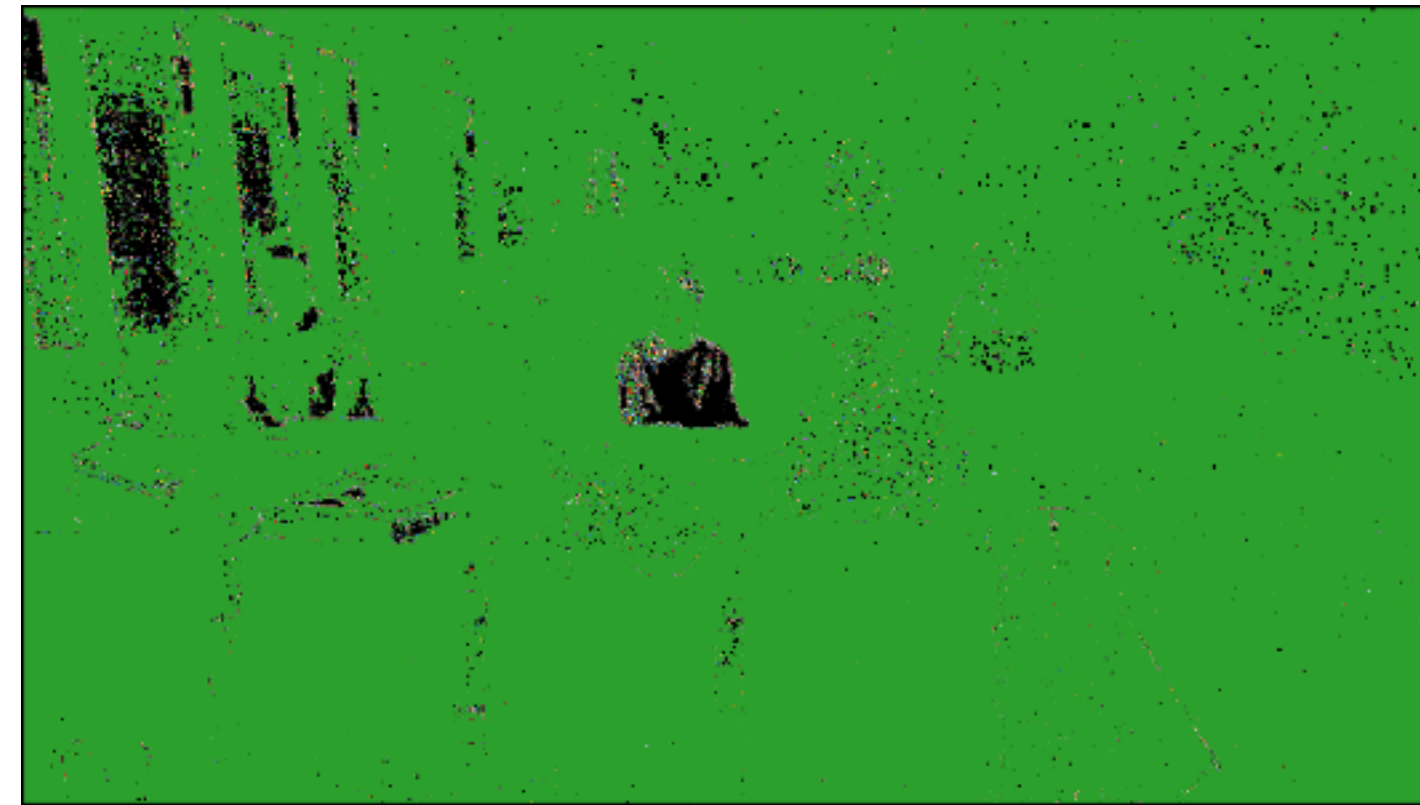
Detection result

# ZERO and double compression

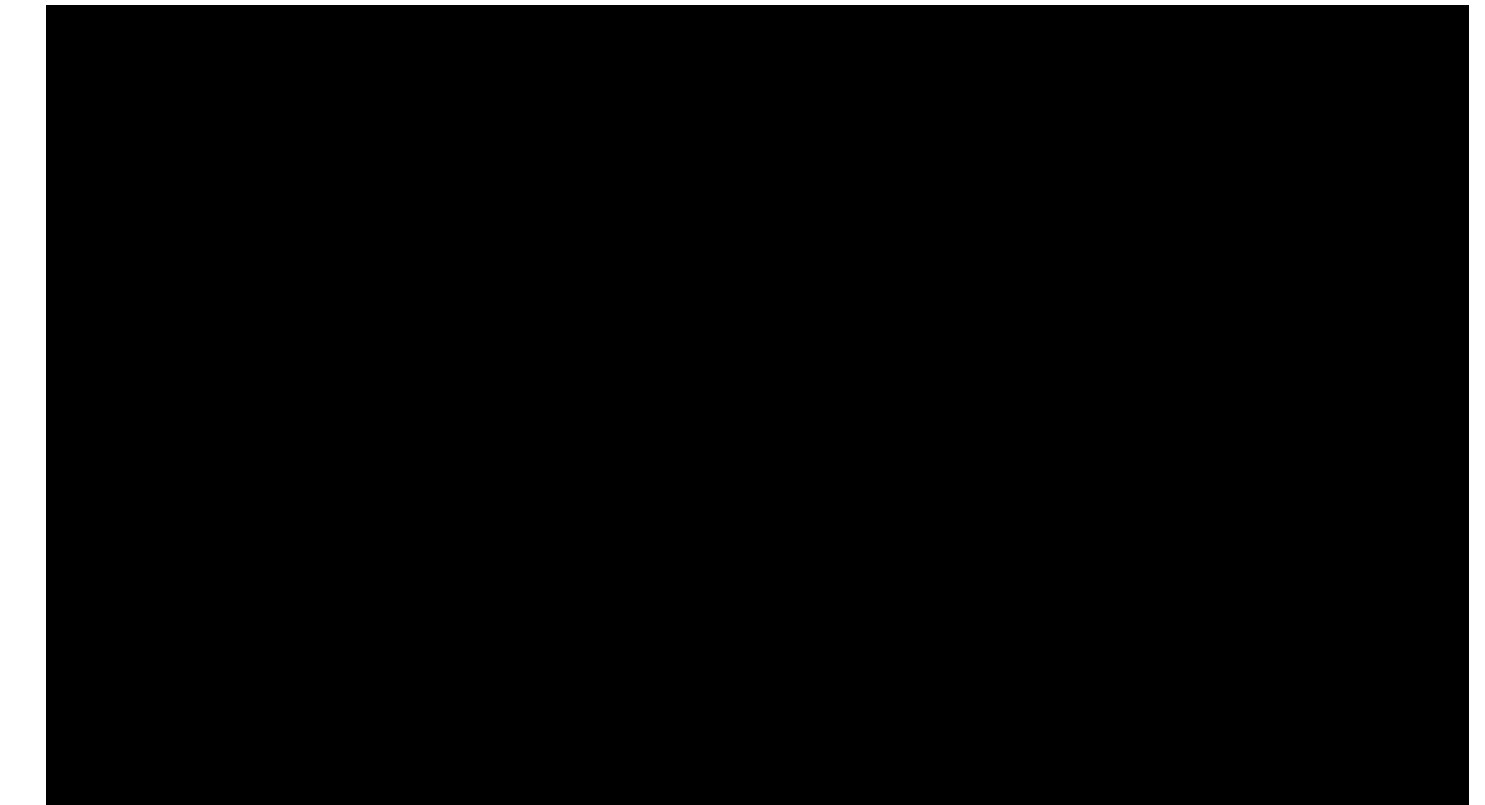
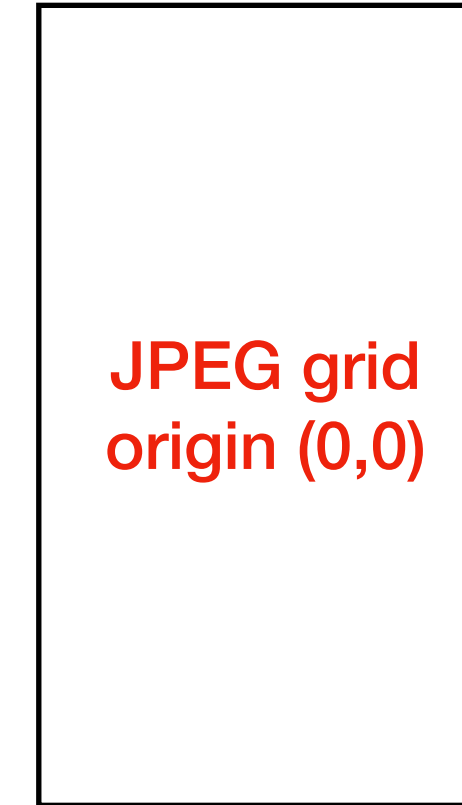
Second compression QF2 << QF1



Forged image



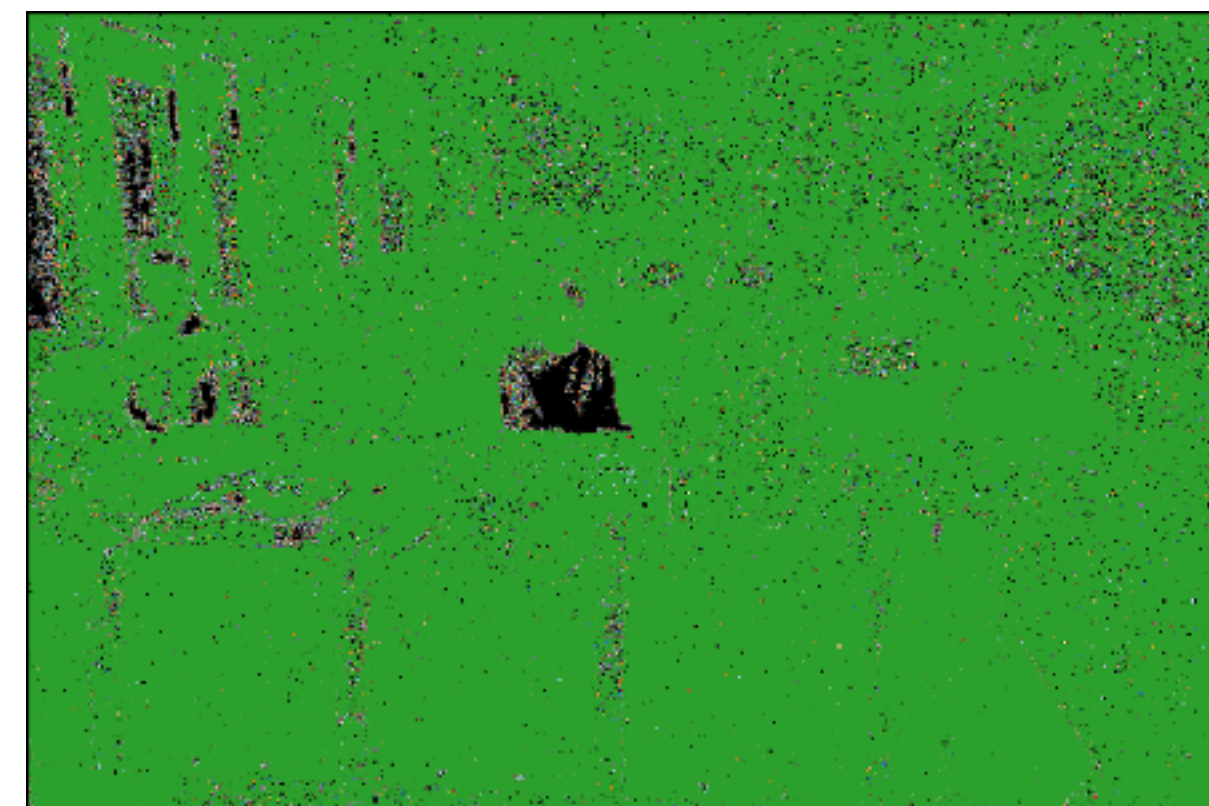
Vote map



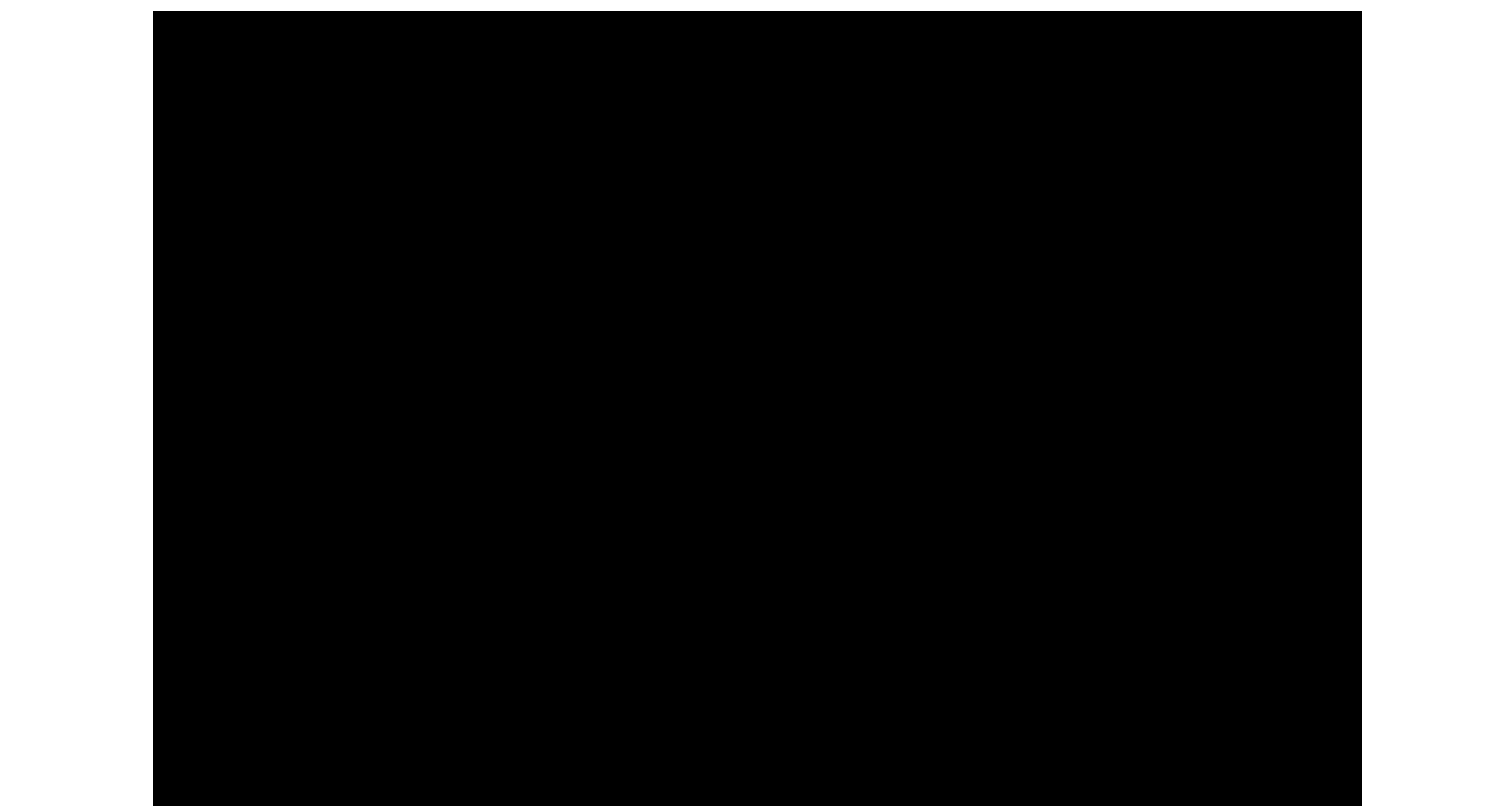
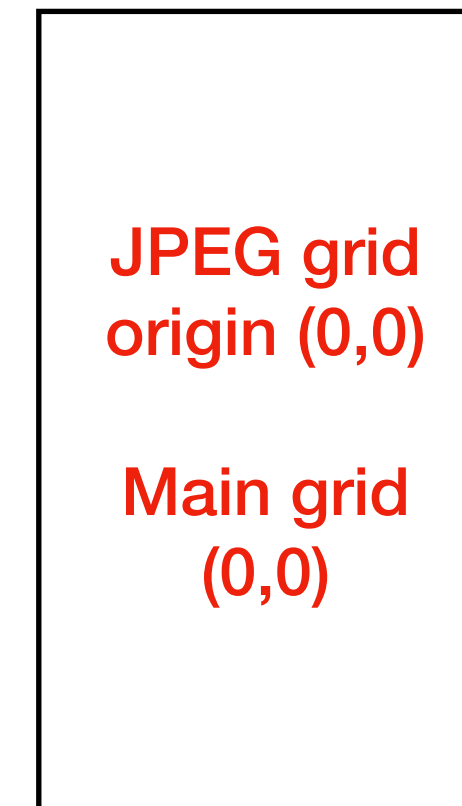
Detection result



Forged and cropped image



Vote map



Detection result

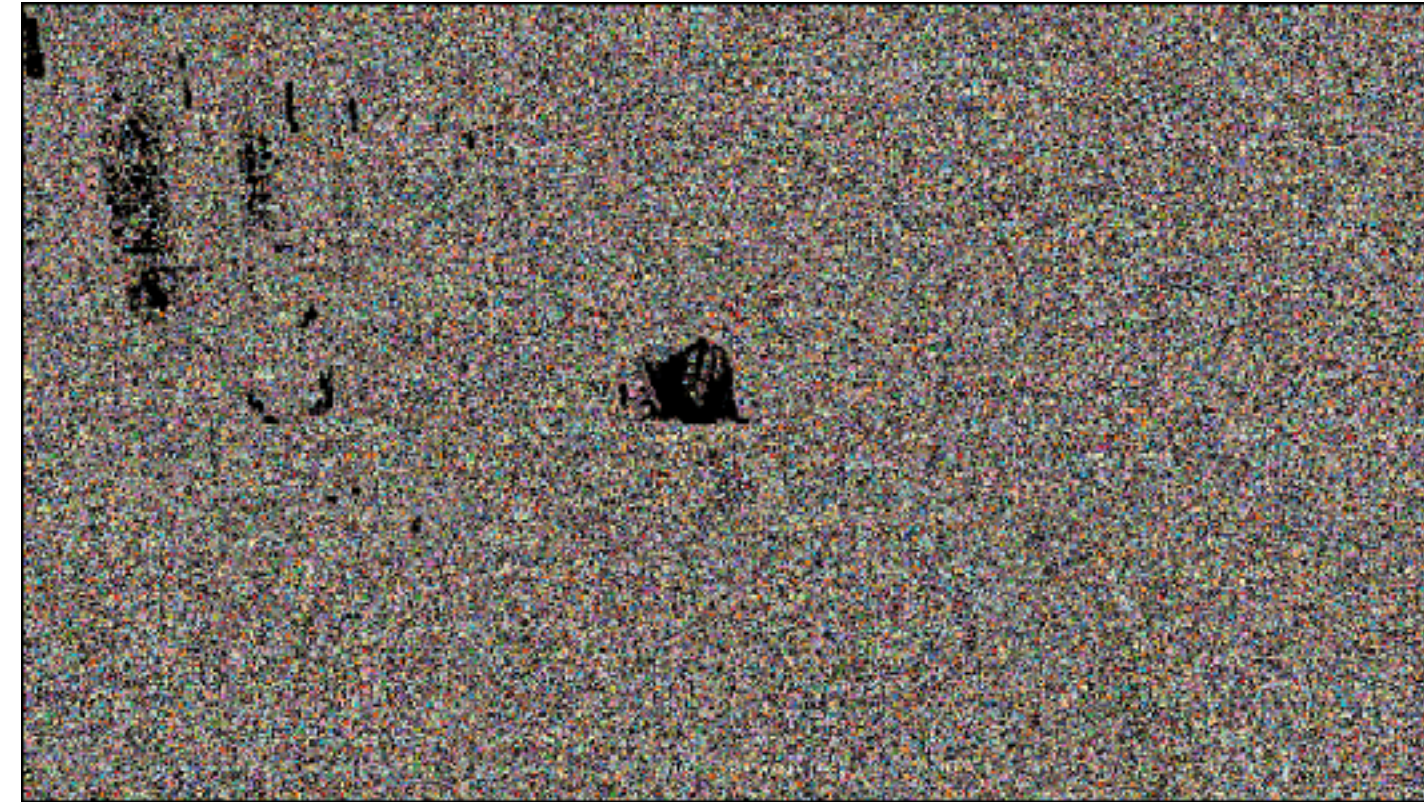
# ZERO and resampling



Downsizing at 90%

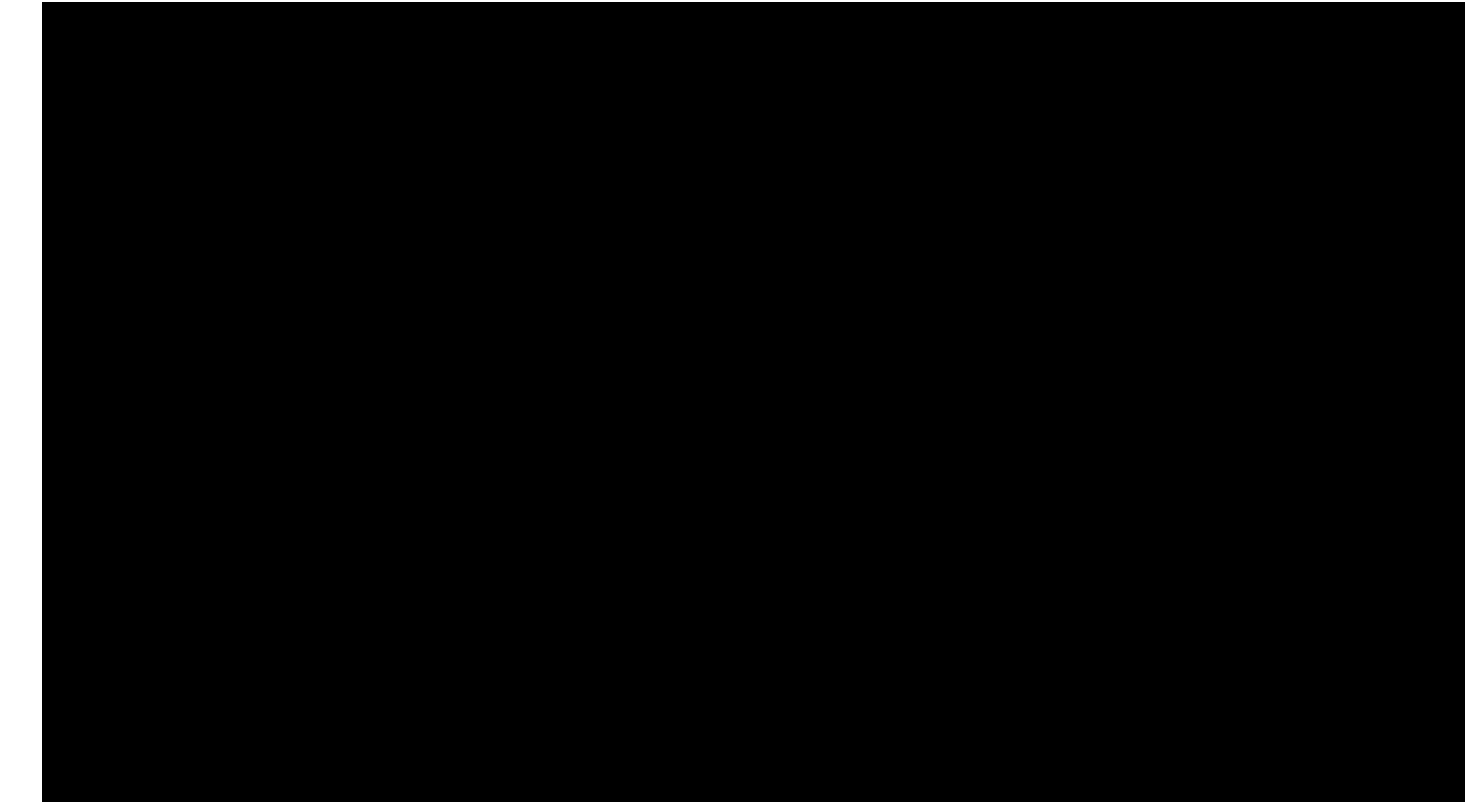


Forged and resampled image



Vote map

No JPEG grid  
detection



Detection result

Rotation

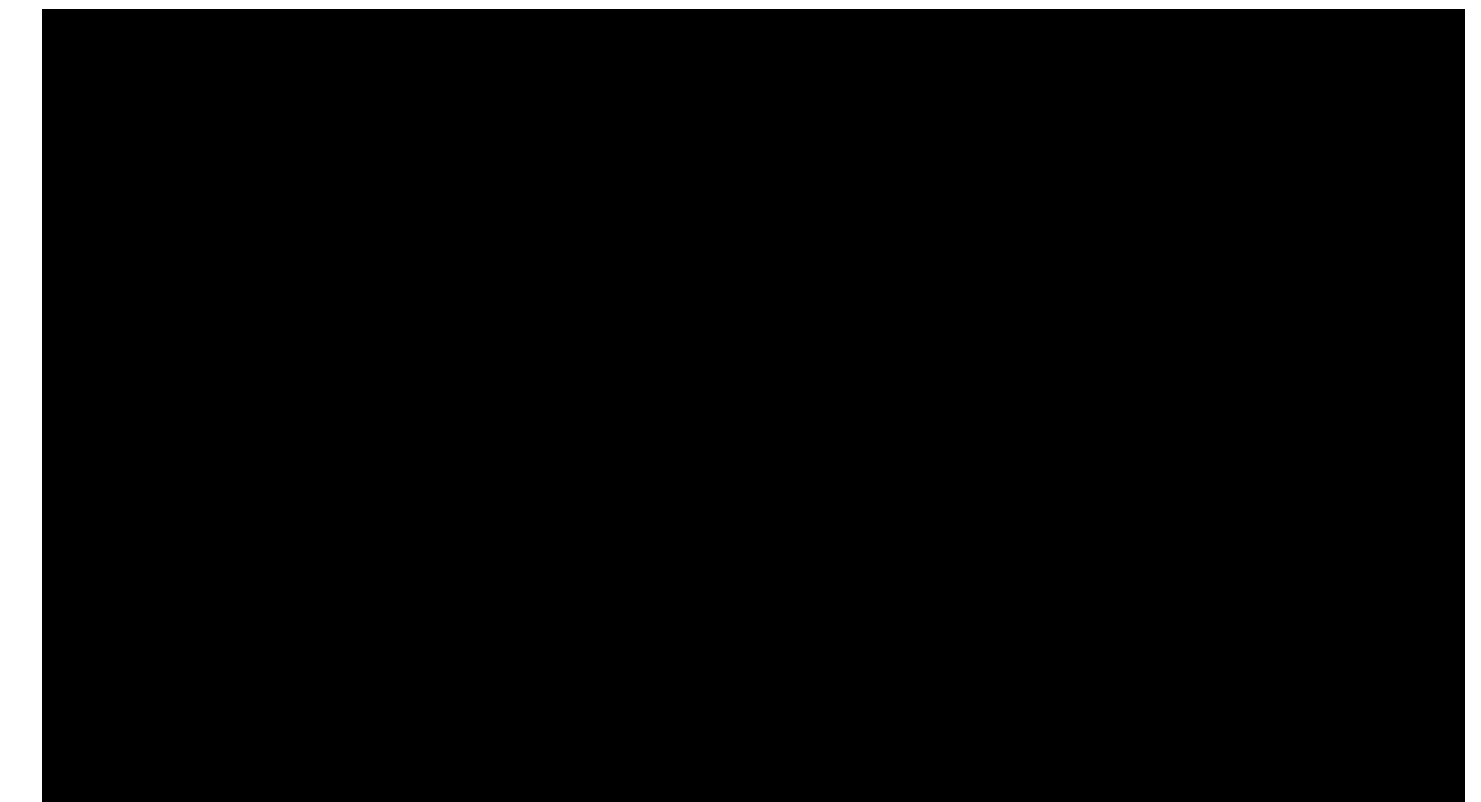


Forged and rotated image



Vote map

No JPEG grid  
detection



Detection result

# ZERO and natural limitations

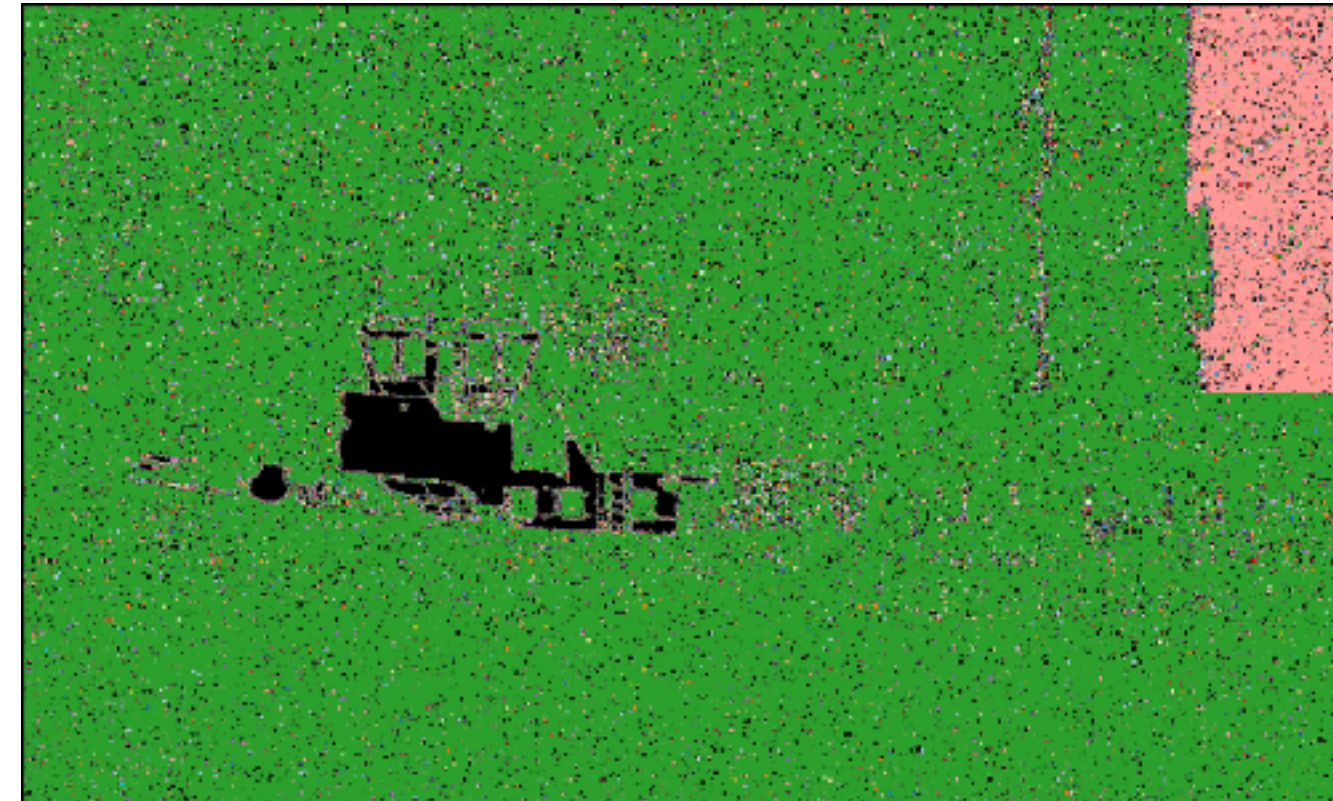
Grid in the correct position with probability 1/64 for a copy-move



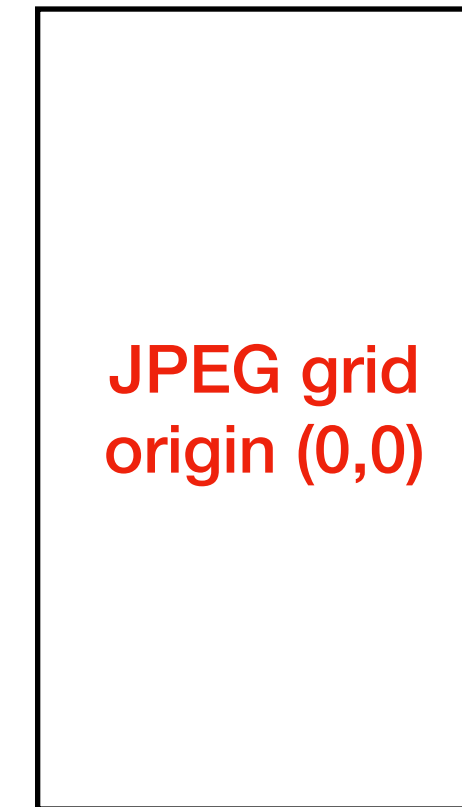
Forged image from FAU dataset.



Ground truth



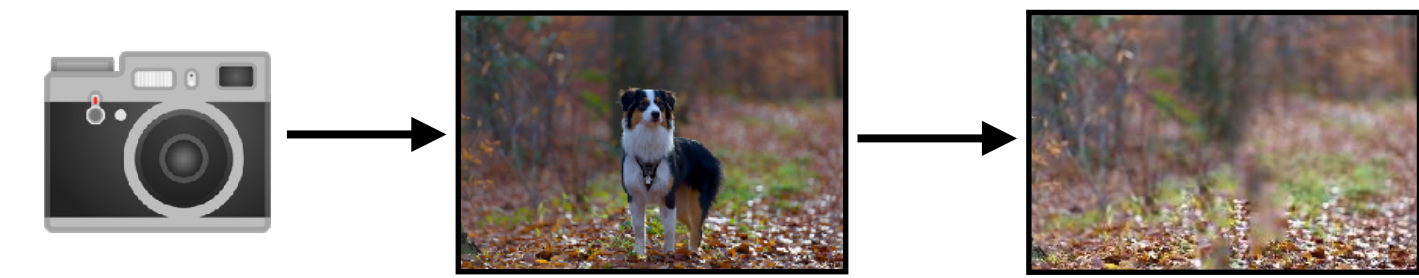
Vote map



Detection result

# ZERO and natural limitations

Uncompressed images



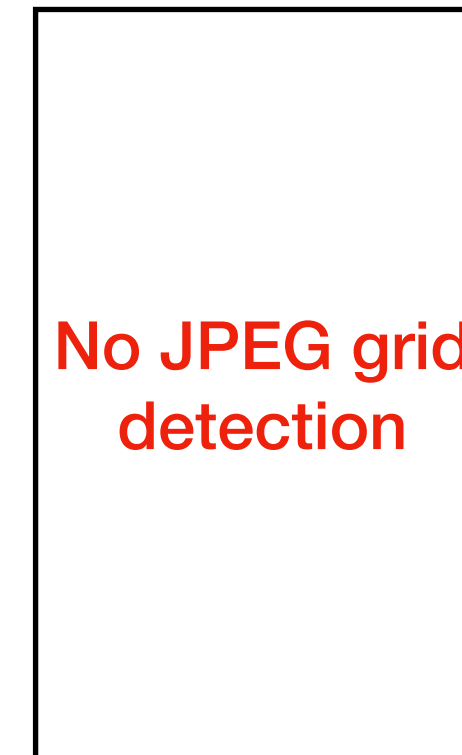
Forged image from Korus dataset.



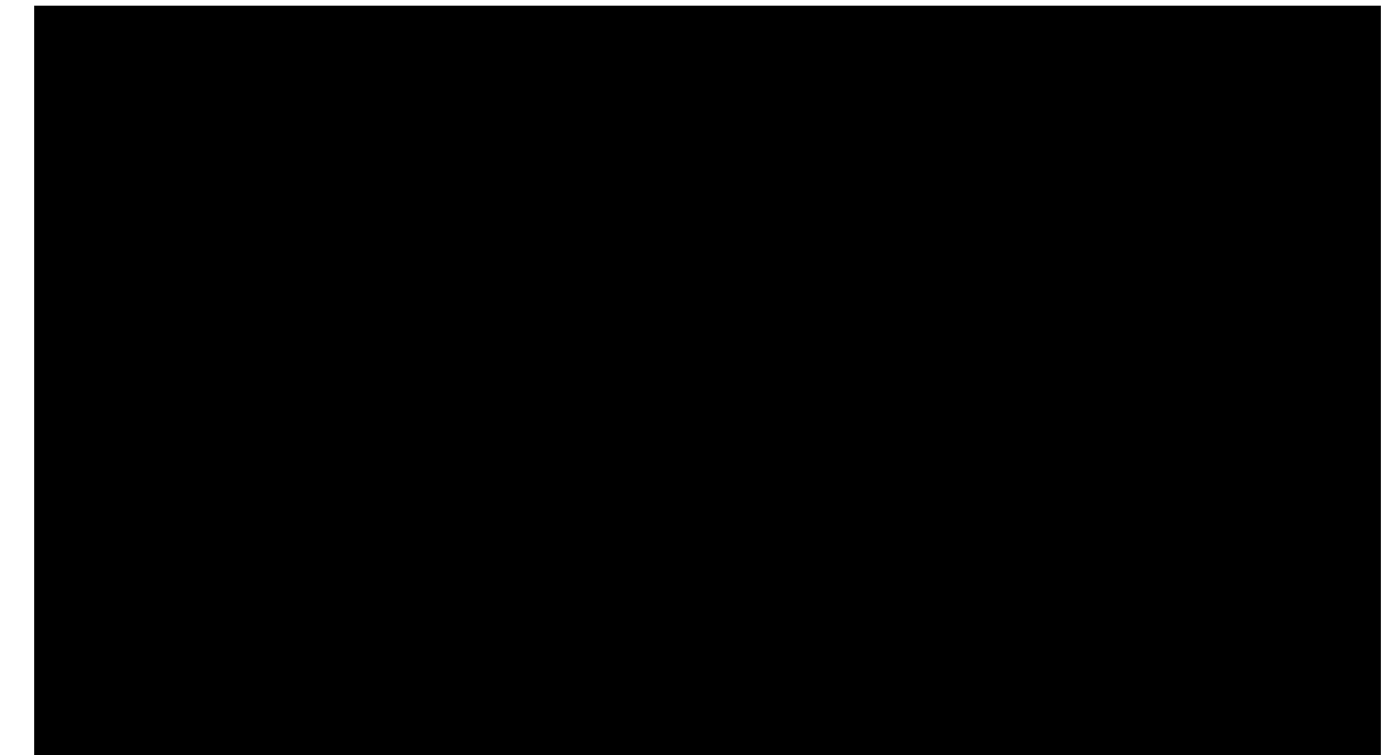
Ground truth



Vote map



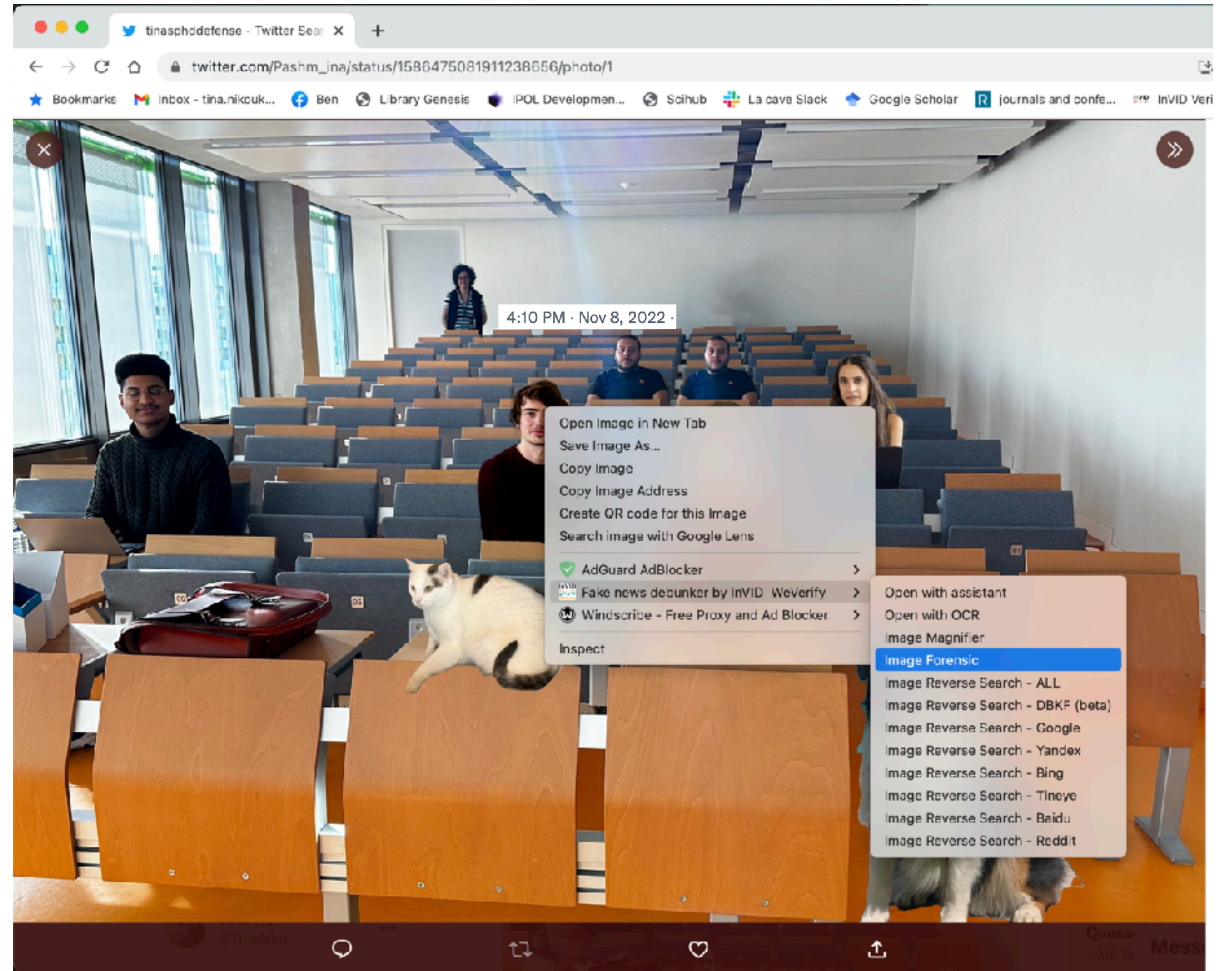
No JPEG grid  
detection



Detection result

# First New reflexes to have

- Look at the image.
- Analyse the image file, but there are no EXIF metadata!
- Perform image reverse search on the Internet, to identify the image's source.
- Use forensic tools.
- Add InVID-WeVerify plug-in to your browser to save time and be more efficient in fact-checking!



Right-click on the image to use the InVID-WeVerify plug-in from AFP news agency.

# An image forensic tool tailed for daily use

**Forensic**

It provides an enhanced toolkit to detect image forgeries

⚠️ This enhanced forensic toolkit aims to help you detect alterations in manipulated images. You should avoid using it with screenshots, scanned images of documents, or juxtaposed images that are in fact altered images. More filters are highlighting the same zone, more suspicious is that particular area of the image. Please take into account that forensic filters are outlining any digital signal alteration and not only semantically manipulated artefacts (which means false positives are possible). Some complex textures or excess of luminance may also alter the signal without any manipulation intention. Whenever possible, use systematically the best image resolution available (even by searching through similarity for higher resolution identical images).

**Analysed image** [NEW IMAGE](#)

**Filters**

COMPRESSION TRACES DEEP LEARNING CLONING

Zero GHST CAGI

Double Quantization DCT BLOCK

No detection  Detection

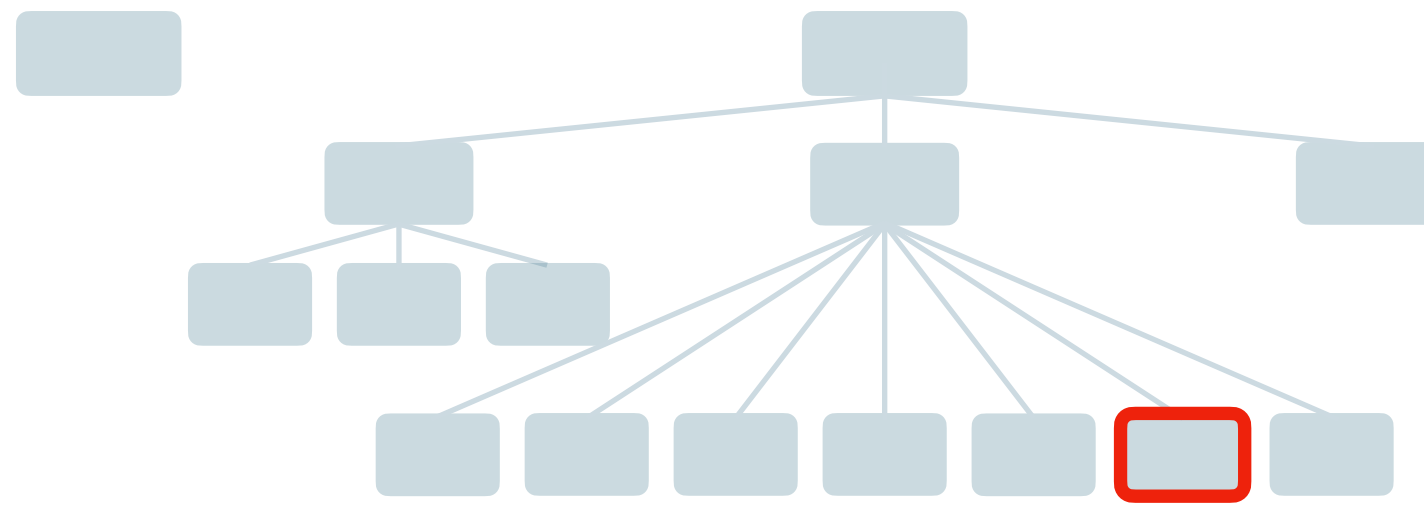
👉 Mouse over the filters to see a transparent mask with the results on the image.

- 📌 These filters detect anomalies in previous processes of creation and compression of the image. If a new element has been removed or added to the image, it can be detected if it has a different compression than the rest of the image.
- ✅ Combination of multiple filters that outline in white or green-light a common area of the image.
- ❌ Complex textures of an object or saturated areas of the photo can generate false positives.

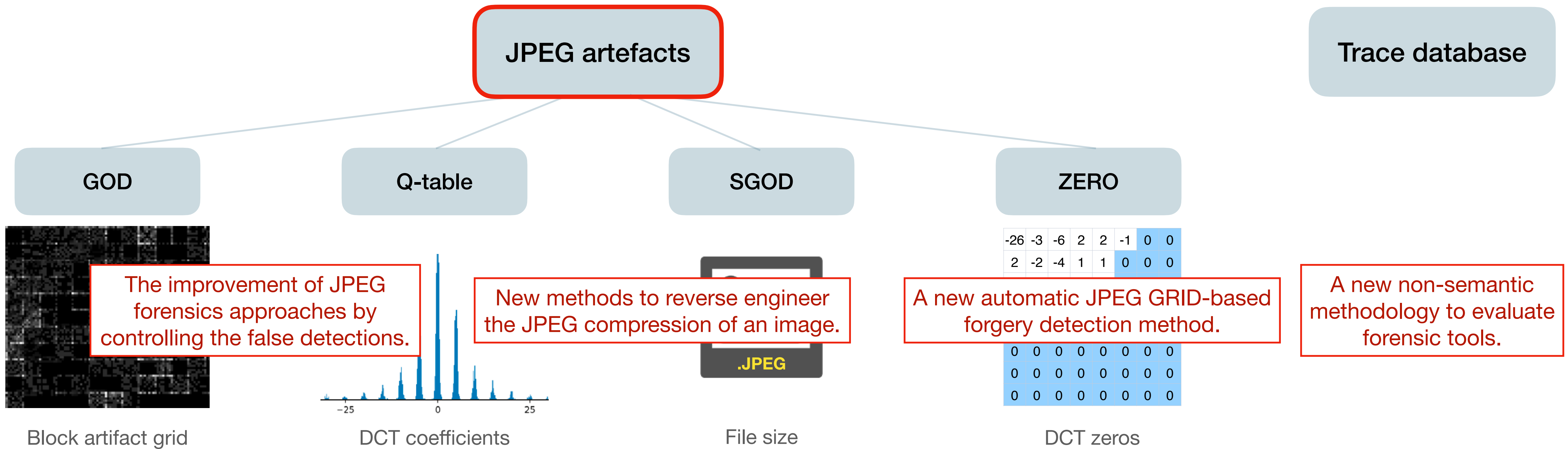
[Feedback](#)

InVID-WeVerify plug-in from AFP news agency.

# Summary of contributions



An overview on image forensics





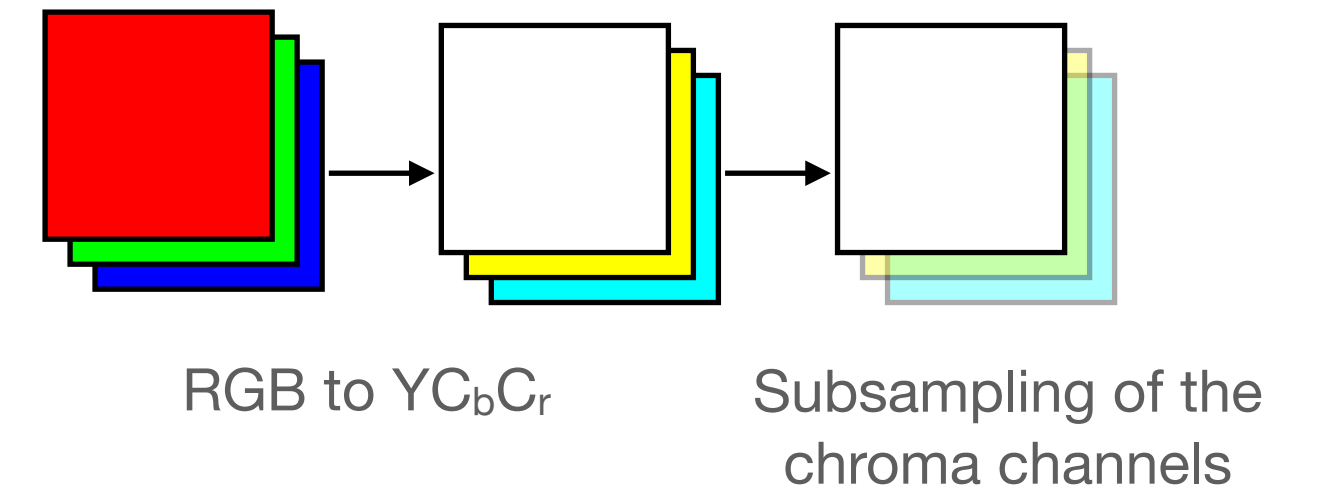
# Future lines of research

## Short-term

- Adapting the existing methods to the **chroma channels**.
- Improving the robustness of the local **missing grid** detection.
- Creating a new automatic JPEG QUALITY-based forgery detection method.
- Adding **post-processing operations** and **new datasets** to the Trace database.

## Long-term

- New/other **compression algorithms** for images and video.
- Fake images generated by **text-to-image** diffusion models!



# Future lines of research

## Short-term

- Adapting the existing methods to the **chroma channels**.
- Improving the robustness of the local **missing grid** detection.
- Creating a new automatic JPEG QUALITY-based forgery detection method.
- Adding **post-processing operations** and **datasets** to the Trace database.

## Long-term

- New/other **compression algorithms** for images and video.
- Fake images generated by **text-to-image** diffusion models!



# The future of image forgery with diffusion models



Crop of an original JPEG compressed image

“students in the classroom”

“Taylor Swift”

“classroom”

“Santa Claus using a laptop”

Text



Mask

# The future of image forgery with diffusion models



Crop of an original JPEG compressed image

“students in the classroom”

“Taylor Swift”

“classroom”

“Santa Claus using a laptop”

Text

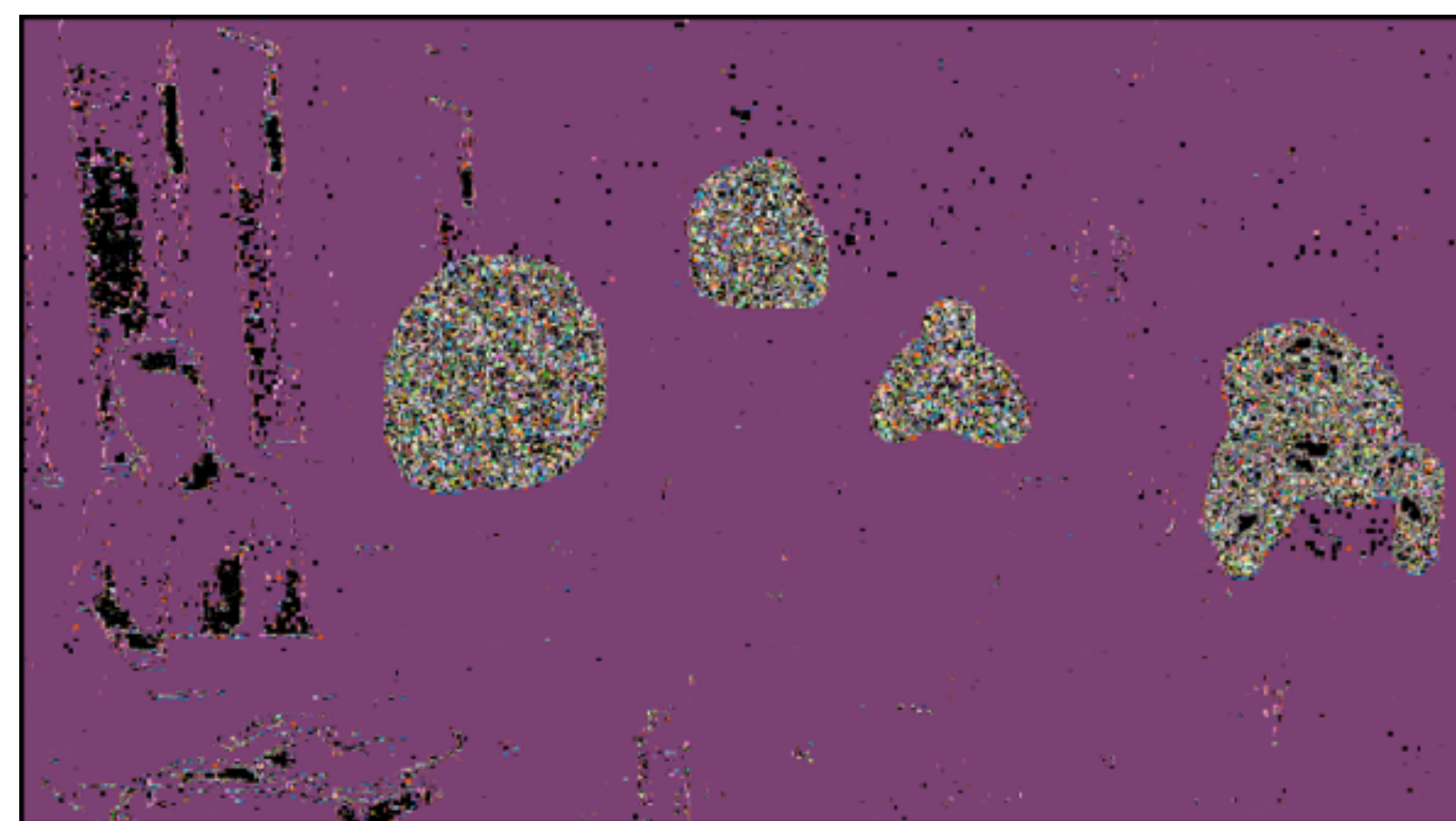


Mask

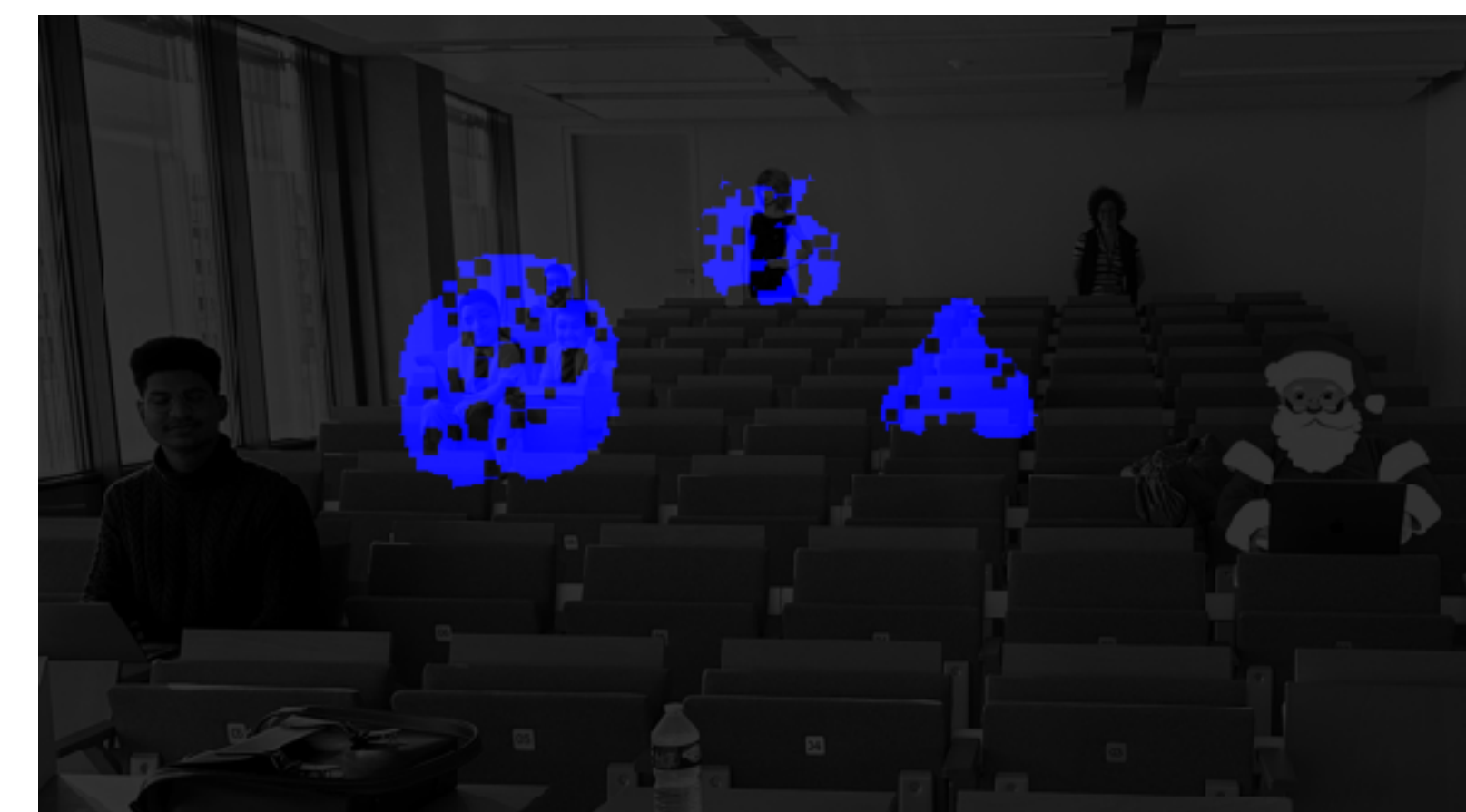


Forged image by stable diffusion v1.5

[R. Rombach et al. 2022]



Vote map



Detection result

# From zero to research, projects, teaching, popular science

CVF This CVPR Workshop paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the accepted version; the final published version of the proceedings is available on IEEE Xplore.

## JPEG Grid Detection based on the Number of DCT Zeros and its Application to Automatic and Localized Forgery Detection

T. Nikiengkheh, J. Anger, T. Ebret, M. Colom, J.M. Morel, R. Grompone von Gioi  
 CMLA, CNRS, ENS Paris-Saclay, Université Paris-Saclay  
 {t.nikiengkheh,anger,ebret,colom,morel,grompone}@cmla.ens-csaclay.fr

### Abstract

This work proposes a novel method for detecting JPEG compression, as well as its grid origin, based on counting the number of zeros in the DCT of  $8 \times 8$  blocks. When applied locally, the same method can be used to detect grid alignment abnormalities. It therefore detects local image forgeries such as copy-paste. The algorithm includes a structural validation step which gives theoretical guarantees on the number of false alarms and provides secure guarantees for forgeries detection. The performance of the proposed method is illustrated with both quantitative and visual results from well-known image databases and comparisons with state-of-the-art methods.

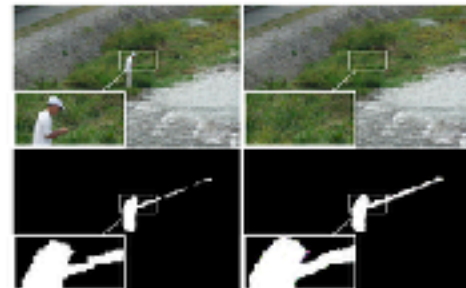


Figure 1. Result of the proposed method, from top left to bottom right: original image, forged image (most of the method), detected forgeries, ground-truth. The fisherman and the fishing rod have been removed. The resulting detected mask does not require visual interpretation.

### 1. Introduction

Image tampering is currently used massively on the web and continuously feeds fake news [12]. This issue has become important since digital image manipulation tools are available to the general public. Some social networks even allow to edit images and videos directly online. Since these platforms (Facebook, Instagram, Snapchat, etc.) share an interest for confidentiality reasons, there is a crucial need for journalists and news-media producers to have access to tools for detecting forgeries from the image itself. Several such tools are readily available online, Fotoforensik, Revealer[6], Glance and Revealer, for instance. These tools provide a number of “tampering localization maps” [26] in the form of so-called image Aesop maps revealing suspicious alterations. However, those localization maps work only as mere enhancers and do not provide any solid proof of forgery. The result needs to be analyzed with care and interpreted by experts. We attempt here to fill in this gap by producing an automatic method using the tampering traces left in the image and taking an automatic decision about forgeries and their precise

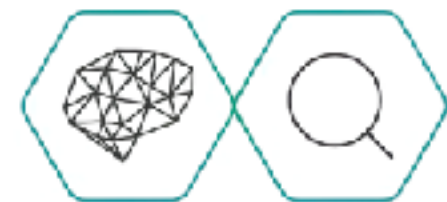
location. Indeed, even in absence of image metadata, much information about the history of the image is still present in the image itself. Many processes undergone by an image leave invisible but detectable traces all over the image. Following [7, 8, 10, 11, 12, 15, 30, 39], we address one of the most common processes, namely the JPEG compression. This compression leaves traces in the form of  $8 \times 8$  pixels blocking artifacts that produce a grid over the image.

Here we propose an automatic and statistically founded method for detecting the grid domain of the image. The proposed method, hereafter referred to as ZERD, performs a global grid detection and then applies it more locally to detect forgeries. Indeed, image splicing generally breaks locally the original grid alignment. A locally detected grid therefore may contradict the global grid and become a reliable cue that a forgery took place. We evaluate ZERD against several state-of-the-art algorithms on publicly available datasets. The detection does not require human experts’ decision as shown in Figure 1, and is associated with theoretical guarantees.

<https://openaccess.thine.org/cvpr2020/46500-0000.html>  
<https://www.github.org> <https://www.robots.ox.ac.uk/~visn/visn.html>



Conferences  
 Publications with code  
 Online demos



Research projects



Master’s level class to journalists  
 Conferences for the general public  
 Workshops during science festivals



Radio shows  
 Interviews  
 Science magazines

# Since November

- Forgery detection
- Deepfake detection
- Fake images generated by diffusion models
- Image restoration
- Video compression analysis
- Satellite imagery
- Scene-text detection (OCR)

Research topics



Research projects



Master's level class to journalists  
Educational platform  
Online journal

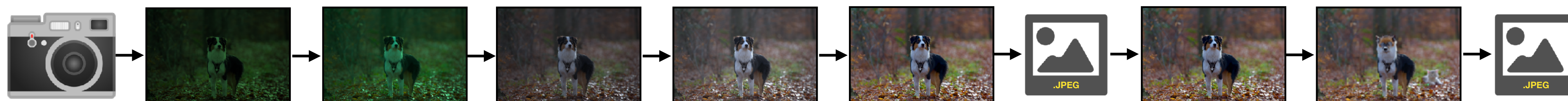


- Middle school, High school, Université Paris-Saclay L3, M1
- Conferences for the general public  
Online videos

# *Thank you for your attention*

## The secret life of JPEG images

Forgery detection using compression traces



These slides are from the thesis defense I  
conducted on November 8th, 2022

<https://nikoukhah.com/tina/phd.html>