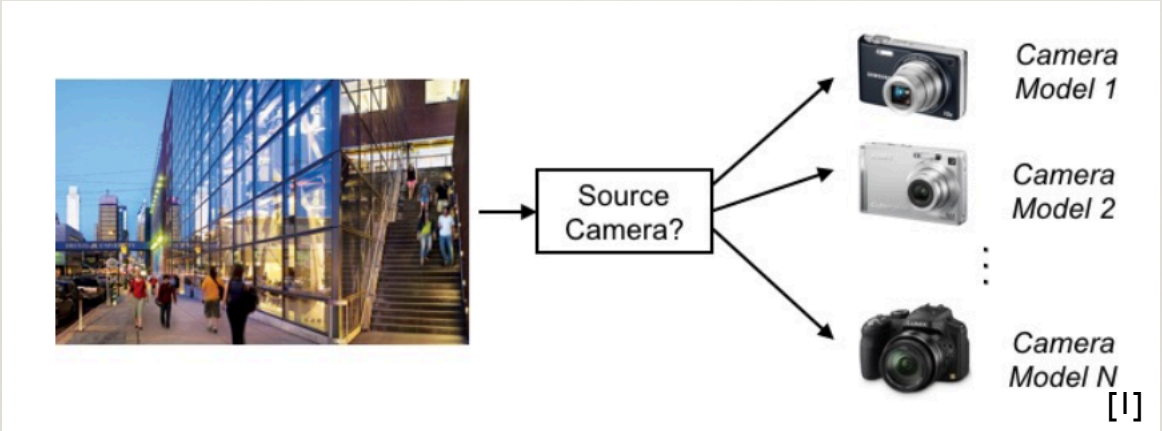
IEEE SIGNAL PROCESSING CUP 2018

MICHAEL GEIGER

ADVISOR: PROFESSOR LUKE DOSIEK

SIGNAL PROCESSING CUP 2018: FORENSIC CAMERA MODEL IDENTIFICATION CHALLENGE

- Required to build a classifier system in order to determine which camera model captured a digital image
  - Cannot rely on image metadata
- Why forensic camera model identification challenge?
  - Criminal investigations, military and defense intelligence, intellectual property theft, etc.



[1]

DESIGN SPECIFICATIONS

- **Open Competition – Part 1**
  - Design signal processing algorithm(s) to extract forensic traces from digital images
  - Design machine learning algorithm(s) to identify a camera model on the basis of these trace
  - Train system on a large dataset of images (10 cameras, 275 images per camera)
- **Open Competition – Part 2**
  - Determine the source camera model of images that have been post processed
- **Open Competition – Data Collection Task**
  - Capture 250 images using a camera model that is not provided in the original dataset
- Algorithms will be implemented using mostly Matlab-based and maybe some Python-based coding schemes

TOP-LEVEL DESIGN: IDENTIFYING CAMERA MODEL SIGNATURES

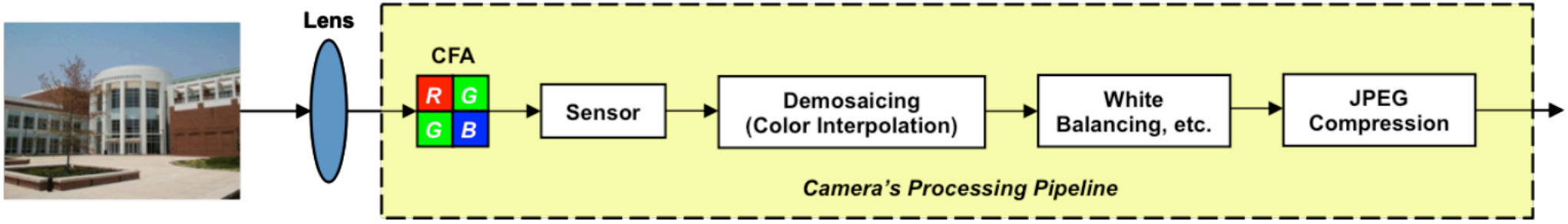


Figure 1: A typical digital camera's internal processing pipeline. [1]

TOP-LEVEL DESIGN: CAMERA ID USING DEMOSAICING SIGNATURES

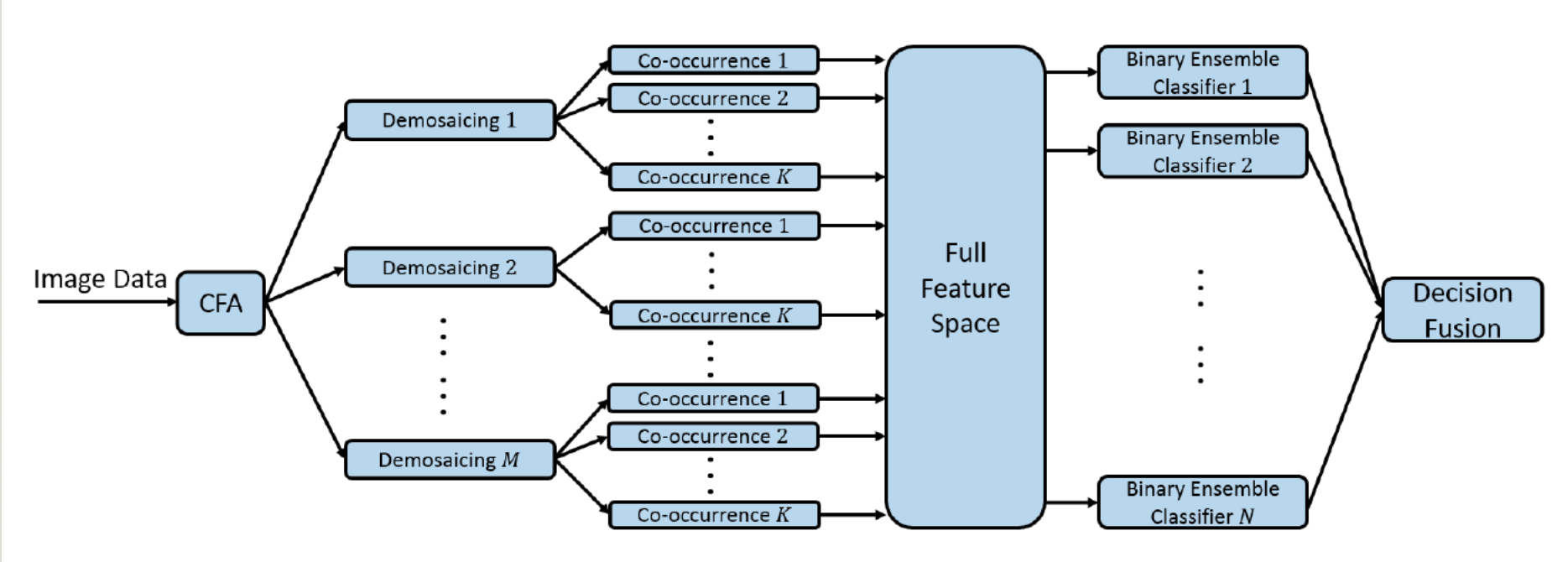


Figure 2: Camera Model Identification Framework [2]

CHARACTERIZATION PLAN: CONFUSION MATRICES

- Separate image database into training database and testing database
- Construct confusion matrices for different combinations of demosaicing/CFA/machine learning algorithms

		True Model											
		1	2	3	4	5	6	7	8	9	10	11	12
Identified Model	1	**99.6%**	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	2	0.0%	**100%**	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%	0.0%	0.0%
	3	0.2%	0.0%	**99.8%**	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.5%
	4	0.2%	0.0%	0.0%	**99.8%**	0.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	5	0.0%	0.0%	0.0%	0.0%	**99.8%**	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
	6	0.0%	0.0%	0.0%	0.0%	0.0%	**99.6%**	0.0%	0.0%	0.3%	0.0%	0.0%	0.0%
	7	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	**99.5%**	0.4%	0.0%	0.0%	0.0%	0.0%
	8	0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	0.5%	**99.3%**	0.0%	0.4%	0.0%	0.0%
	9	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	0.0%	**98.6%**	0.4%	0.1%	0.2%
	10	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.4%	0.3%	**98.3%**	0.1%	0.2%
	11	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%	**98.8%**	1.8%
	12	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.9%	0.8%	**97.3%**

Figure 3: Example Confusion Matrix for 12 Different Camera Models [2]

COMPETITION TIMELINE

- **End of Fall Term**: Construct a functioning camera model identification system
- **January 22**: Open Competition Deadline
  1. Report in the form of IEEE conference paper
  2. Camera model identification results
  3. Data collection task of 250 images
  4. An executable with a Matlab implementation of camera model identification system
     - Produce a text file identifying camera model
- **February 10**: Announcement of Three Finalists
- **April 22-27**: Final Competition at ICASSP 2018 in Seoul, South Korea

ACKNOWLEDGEMENTS

- Professor Luke Dosiek
- Signal Processing Cup Practicum Students

QUESTIONS?

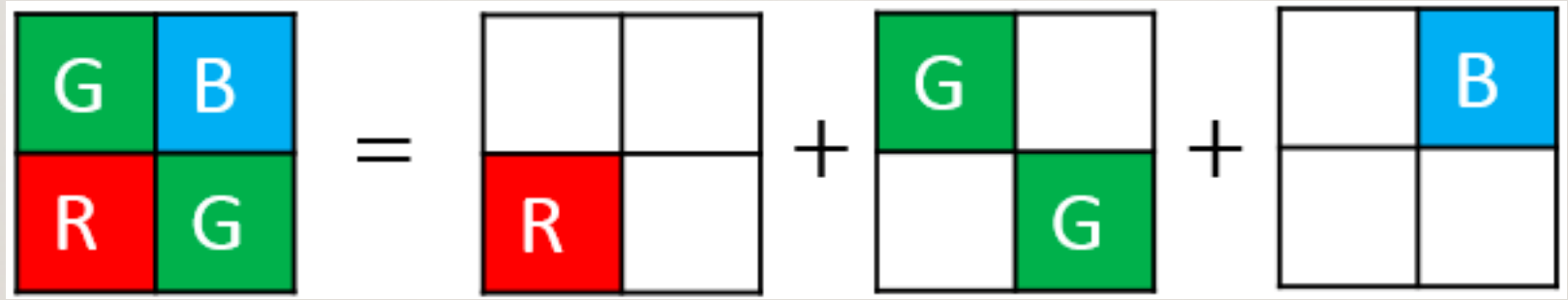
REFERENCES

[1] http://signalprocessingsociety.org/sites/default/files/uploads/get_involved/docs/SPCup_2018_Document.pdf
[2] http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7368573

APPENDIX

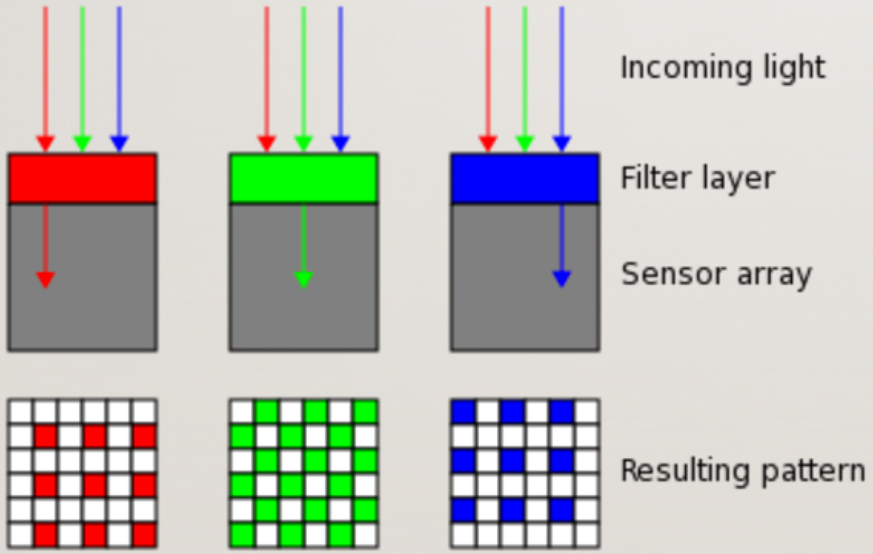
BAYER COLOR FILTER ARRAY (CFA)

- CFA allows only one color component of light to pass through it at each position
  - the sensor records only one color value at each pixel location

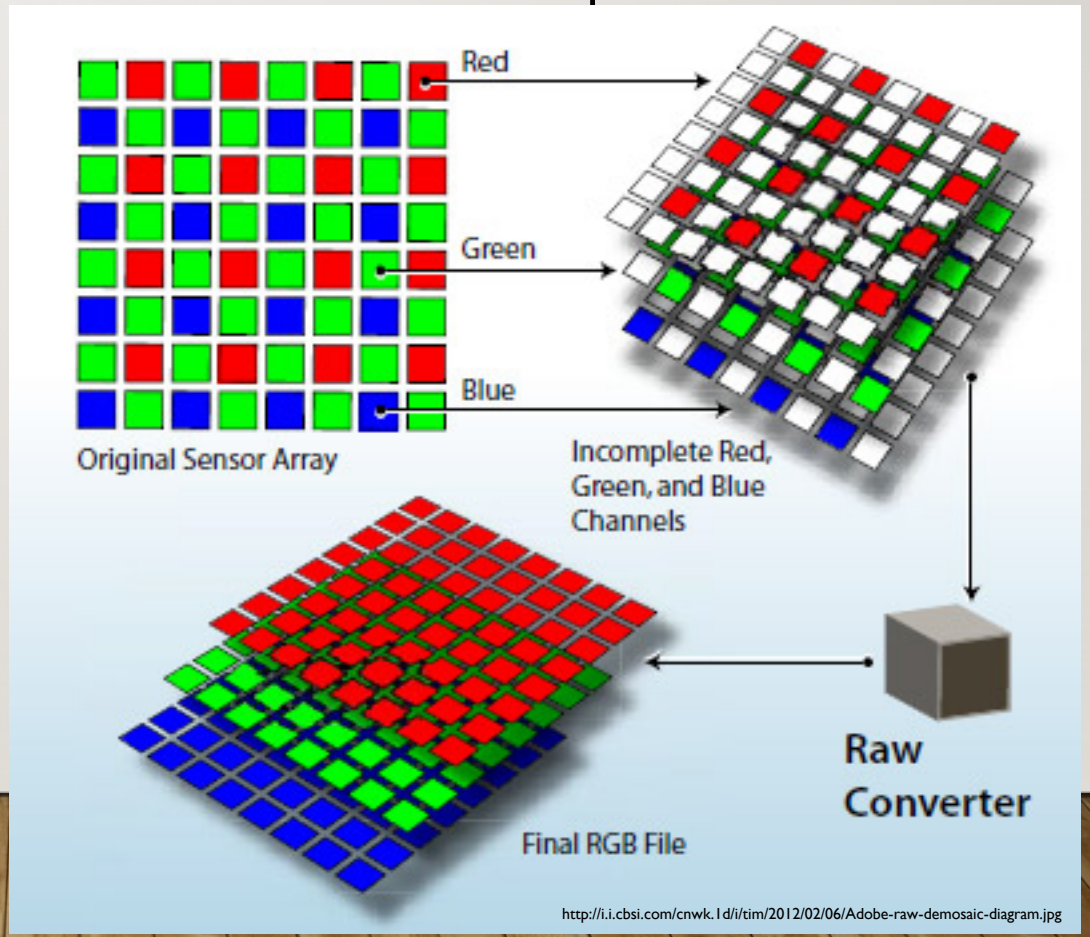


WHAT IS DEMOSAICING?

- The process of interpolating the two unobserved color values at each pixel location



https://rawpedia.rawtherapee.com/images/thumb/1/1c/Bayer_pattern_on_sensor_profile.svg/300px-Bayer_pattern_on_sensor_profile.svg.png



http://i.i.cbsi.com/cnwk.1d/i/tim/2012/02/06/Adobe-raw-demosaic-diagram.jpg

DEMOSAICING ALGORITHMS

- Nearest Neighbor Interpolation
- Bilinear Interpolation
- Smooth Hue Transition Interpolation
- Median-Filtered Bilinear Interpolation
- Gradient-Based Interpolation
- Matlab's Gradient-Corrected Linear Interpolation

CLASSIFIER FRAMEWORK

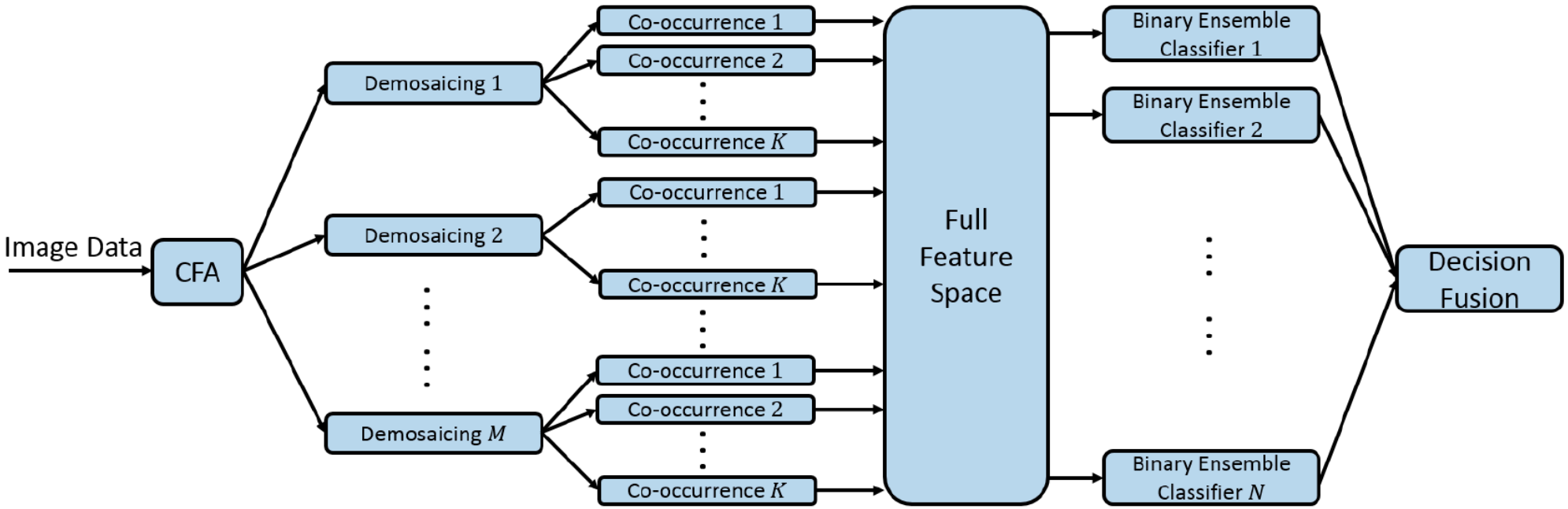


Fig. 3. Architecture of our camera model identification framework.

ORIGINAL VS. RECONSTRUCTED IMAGE EXAMPLE

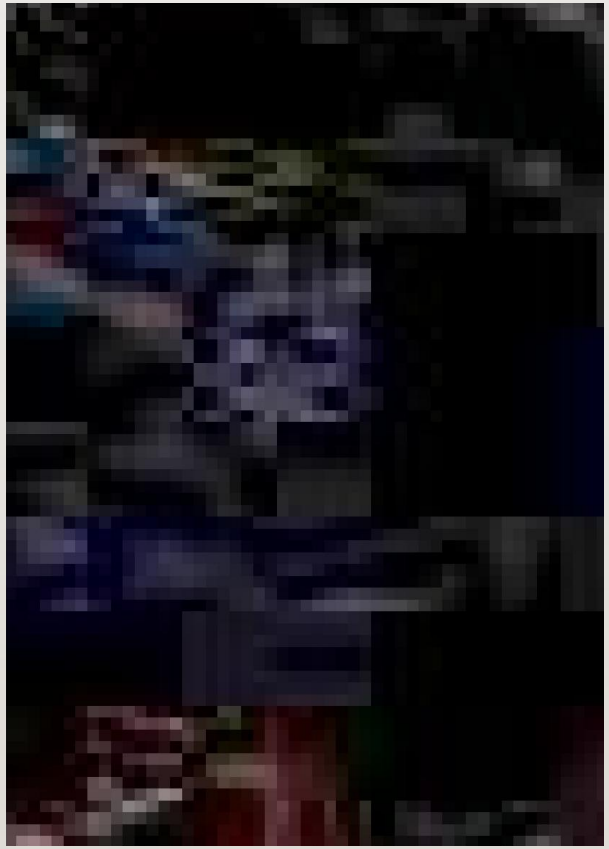
ORIGINAL

RECONSTRUCTED

CONSTRUCTING "ERROR IMAGE"

$$\mathbf{E} = \mathbf{X} - \mathrm{Demos}_{CFA,H}(\mathbf{X})$$

ERROR IMAGE EXAMPLE
(HIGHLY EXAGGERATED)



QUANTIZATION AND TRUNCATION OF "ERROR IMAGE"

$$\mathbf{E} \leftarrow \text{trunc}_T \left(\text{round} \left(\frac{\mathbf{E}}{q} \right) \right)$$

Where **E** is error image, q is quantization variable (q = 2), and T is truncation variable (T = 3).

FUNCTIONAL DECOMPOSITION