Digital Image Processing

Part 3: Affine Transformations

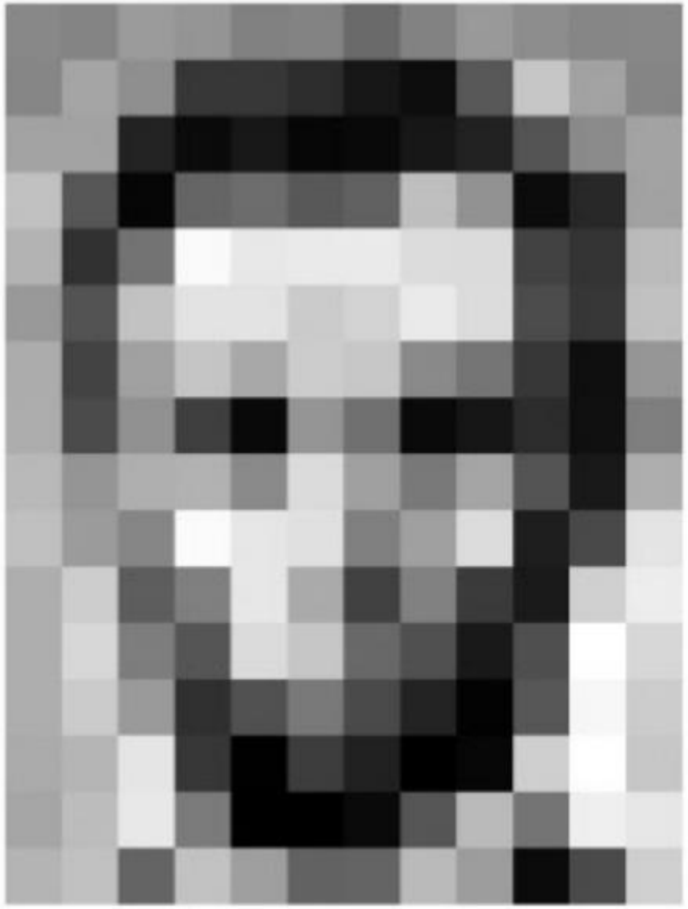
By D.J. Lopez, CCpE, M.Sc.

Overview

- Review of Digital Image Structure
- Linear Transformations and Geometric Transforms
- Noise and Interpolation

Review

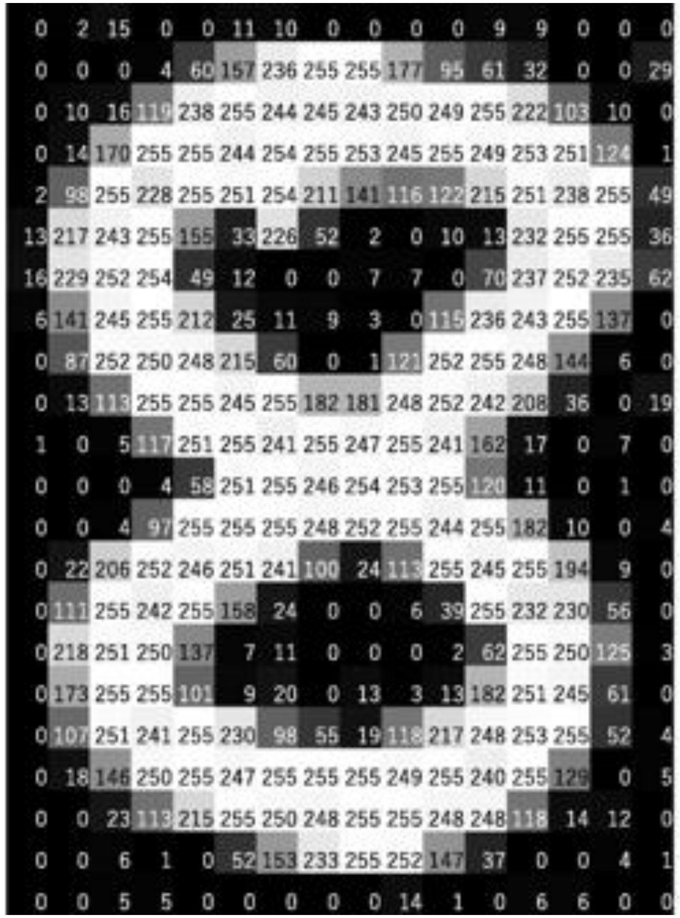
A Grayscale Picture

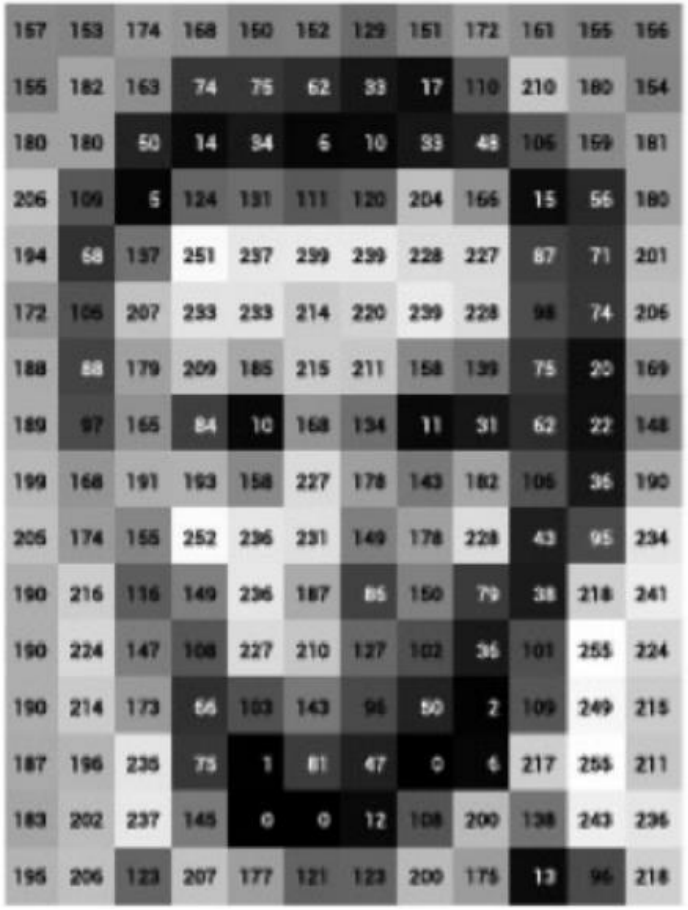


Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9

Melvin Wevers, Thomas Smits, The visual digital turn: Using neural networks to study historical images, *Digital Scholarship in the Humanities*, Volume 35, Issue 1, April 2020, Pages 194–207, https://doi.org/10.1093/llc/fqy085

Two-dimensional signal





Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9

Melvin Wevers, Thomas Smits, The visual digital turn: Using neural networks to study historical images, *Digital Scholarship in the Humanities*, Volume 35, Issue 1, April 2020, Pages 194–207, https://doi.org/10.1093/llc/fqy085

Two-dimensional vector: A Matrix

$X_a =$

0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

Shape: (rows, columns) = (22,16)

$X_b =$

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Shape: (rows, columns) = (16,12)

Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9

Melvin Wevers, Thomas Smits, The visual digital turn: Using neural networks to study historical images, *Digital Scholarship in the Humanities*, Volume 35, Issue 1, April 2020, Pages 194–207, https://doi.org/10.1093/llc/fqy085

Into the Matrix

Channels

Three-dimensional vector: A Tensor

red

green

blue

(480,480)

(480,480)

(480,480)

(480,480, 3)

Linear Transformations

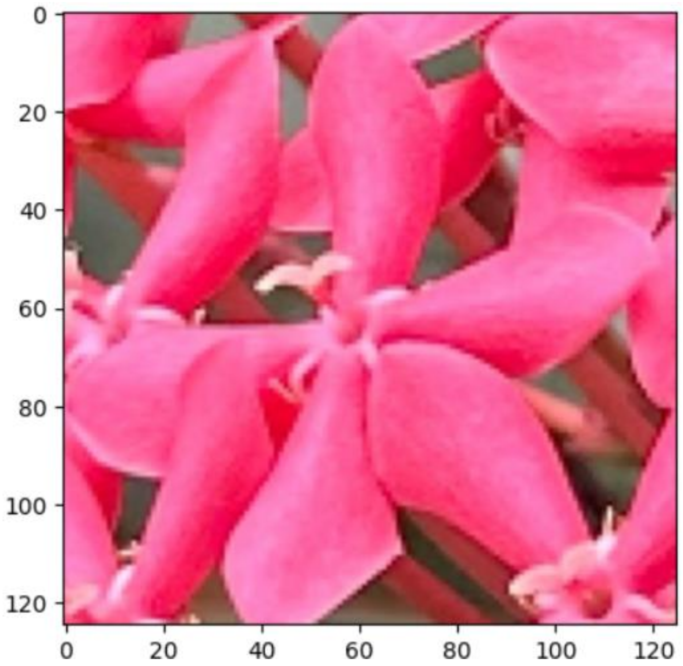
Splicing or Cropping

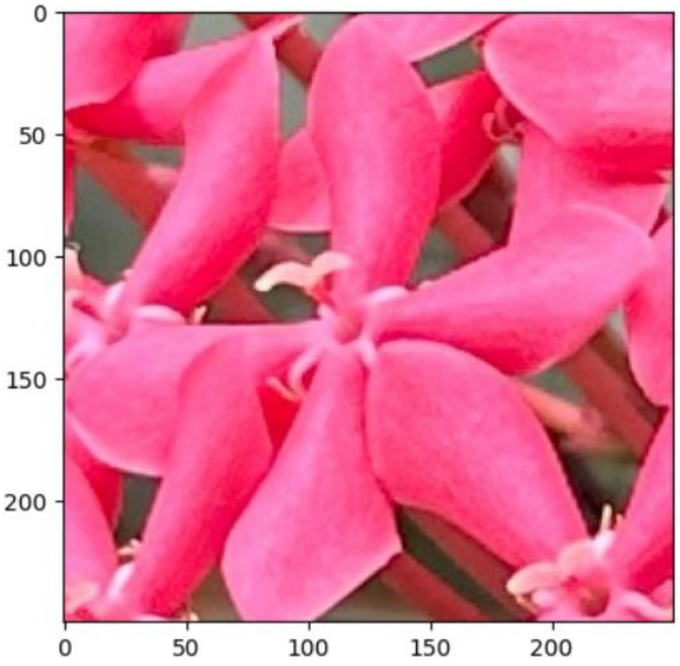
Cropping is a n-dimensional signal splicing technique. The resulting cropped image is always a subset of an original image.

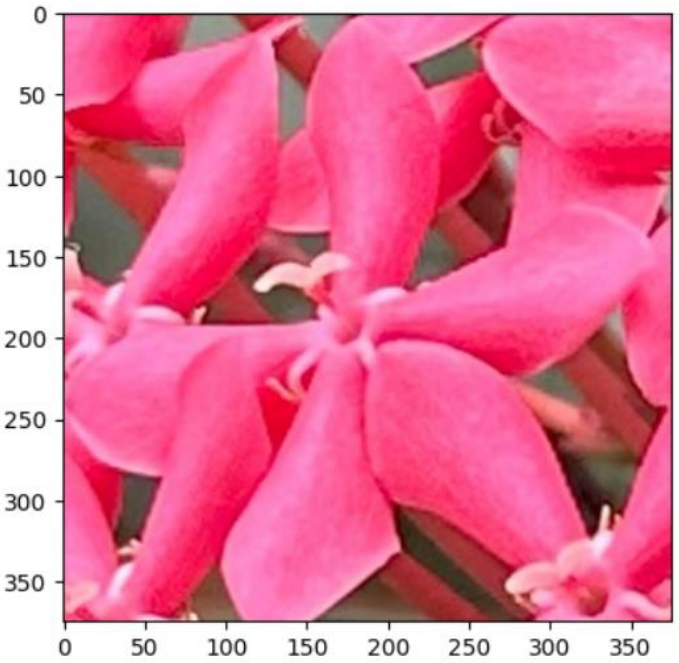
Scaling

Spatial scaling or expansion is the equivalent of upsampling or downsampling of an image not amplification or attenuation. Other terms for image scaling is zooming in most software applications.

Scaling is also subjected by sampling errors which are evident with image **artifacts**

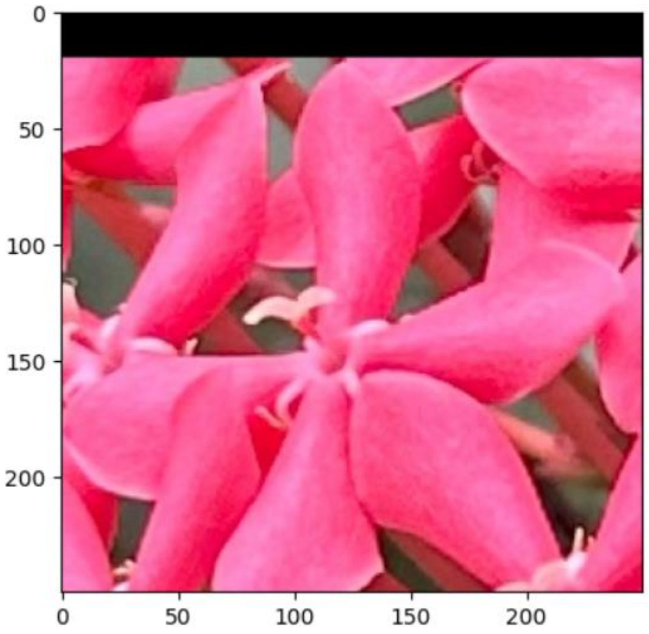


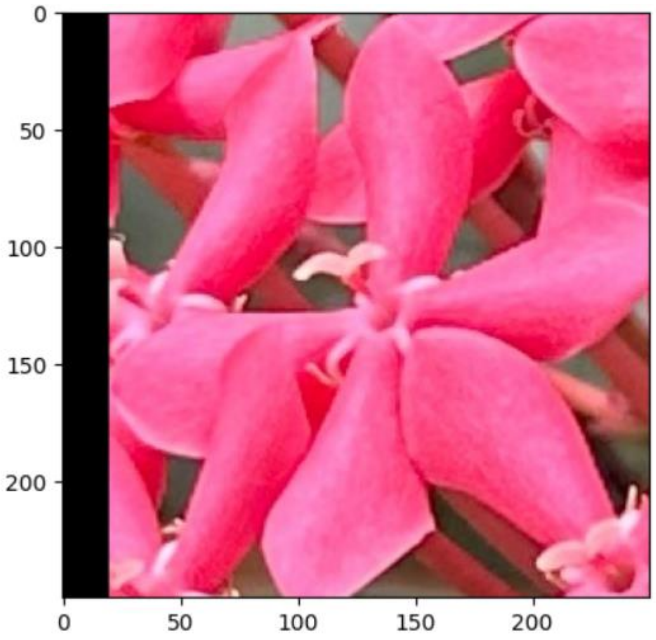


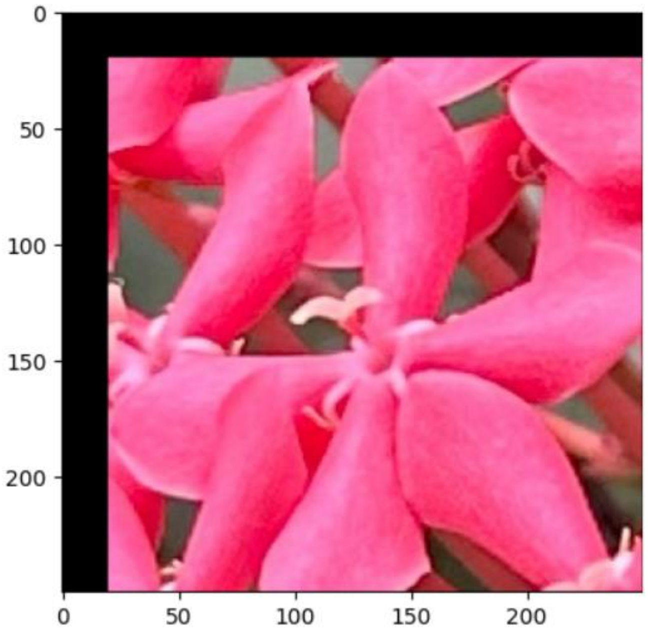


Shifting

Shifting or spatial translation preserves the orientation of the image while shifting all pixels in the same direction and order.

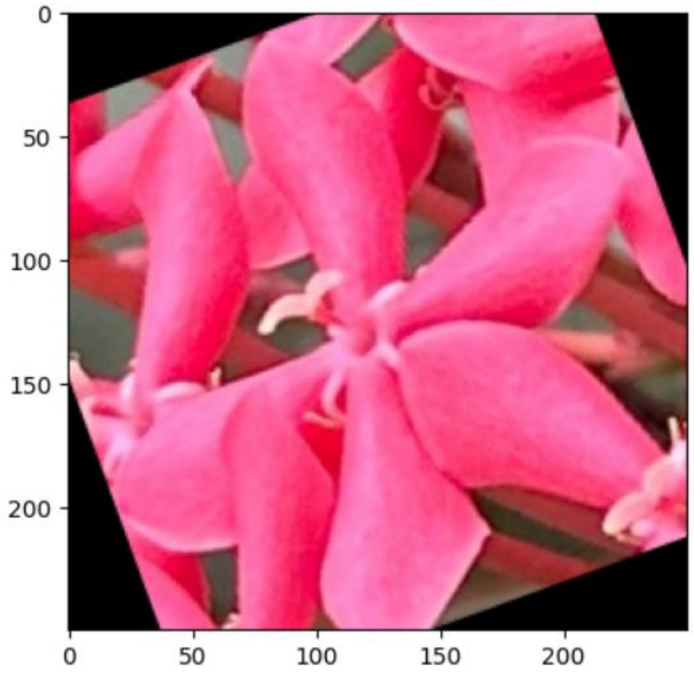


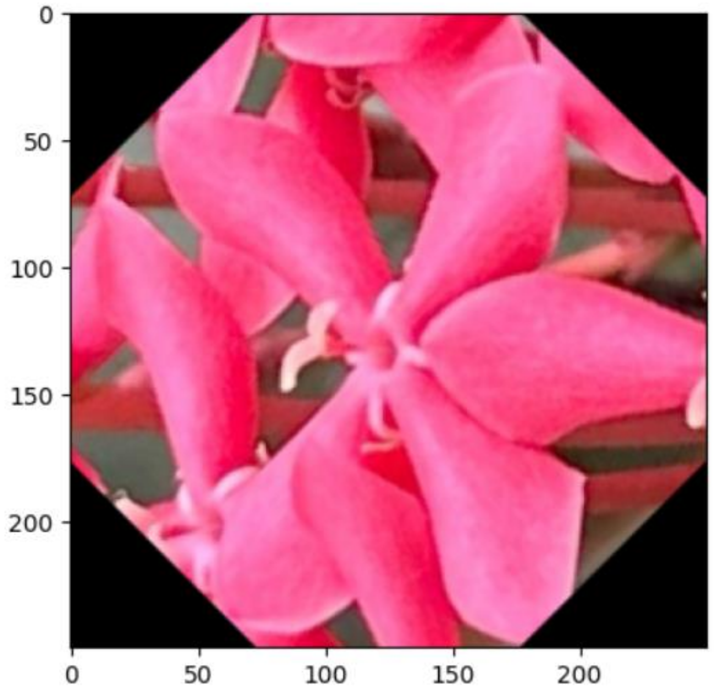


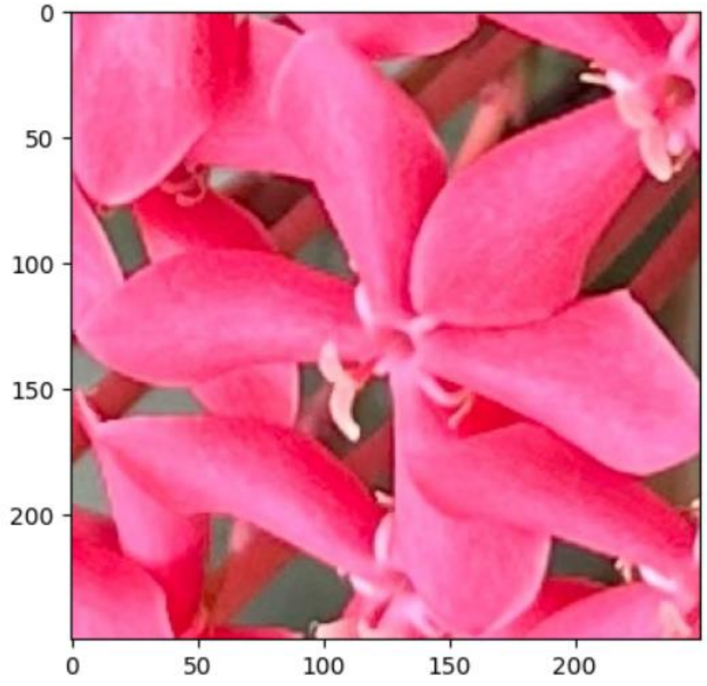


Rotating

Rotation, similar to the rotation linear operator, rotates an image by a certain angle. In OpenCV, the image is rotated while preserving areas not included in the image.

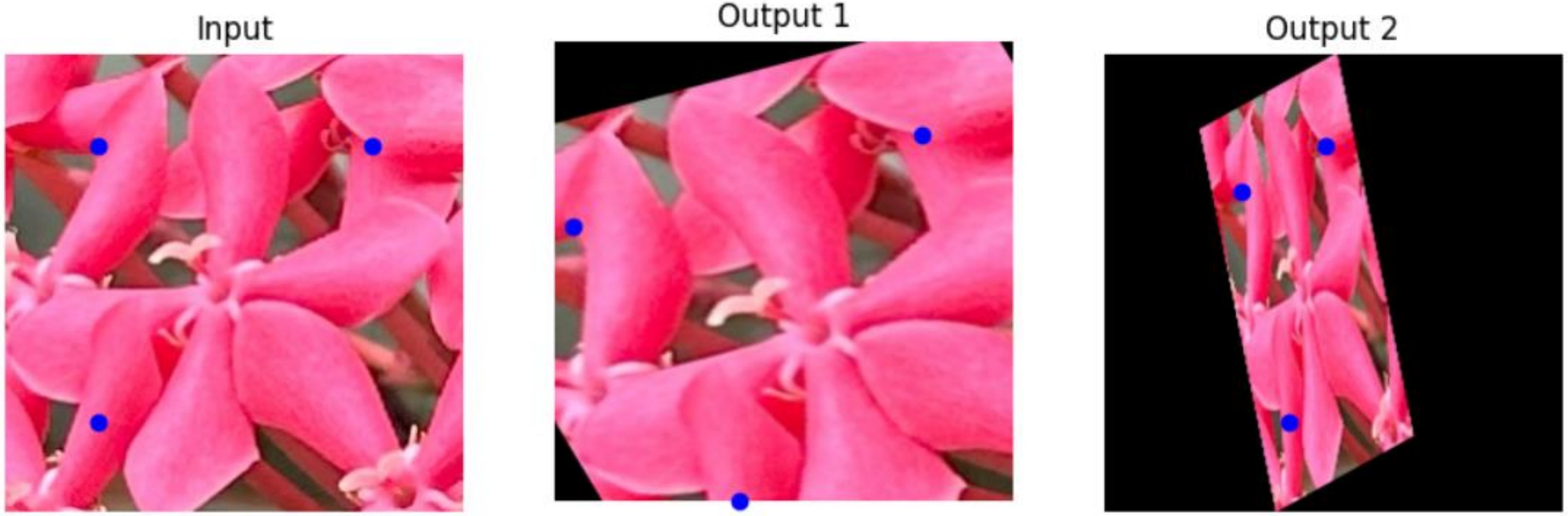






Affine Transformation (Image Warping)

Affine transformations are generalized linear operators which may include shifts, rotations, or scales. The affine transform requires three reference points to determine the entire spatial transformation



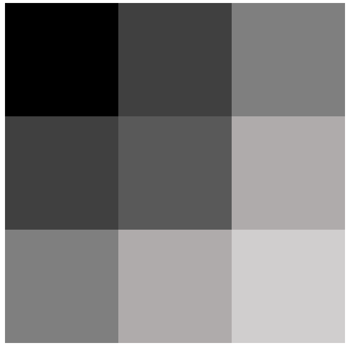
Noise and Interpolation

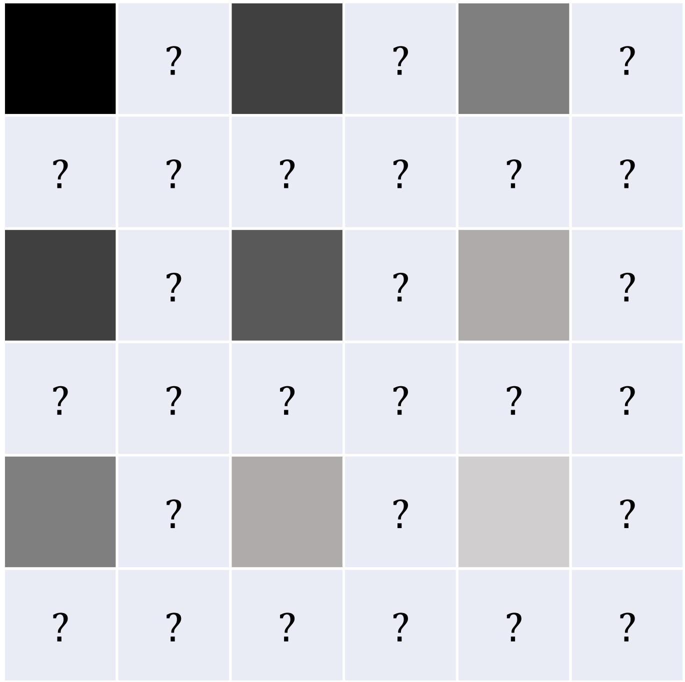
Why interpolate?

Interpolation. A technique determining values in between two points or samples.

Spatial expansion (e.g., elongation, zooming, shifting, and rotating) induces spaces between pixels.

We make use of interpolation to determine the values of the values between the expanded placement of pixels.





Interpolation Techniques

1. Nearest-Neighbor Interpolation
2. Bilinear Interpolation
3. Bicubic Interpolation

Interpolation Techniques

1. Nearest-Neighbor Interpolation

The nearest-neighbor interpolation or point-sampling takes the $n \times n$ neighbors of a point/pixel to adapt (copy) its value.

0	64	127
64	89	172
127	172	175

0	?	64	?	127	?
?	?	?	?	?	?
64	?	89	?	172	?
?	?	?	?	?	?
127	?	172	?	175	?
?	?	?	?	?	?

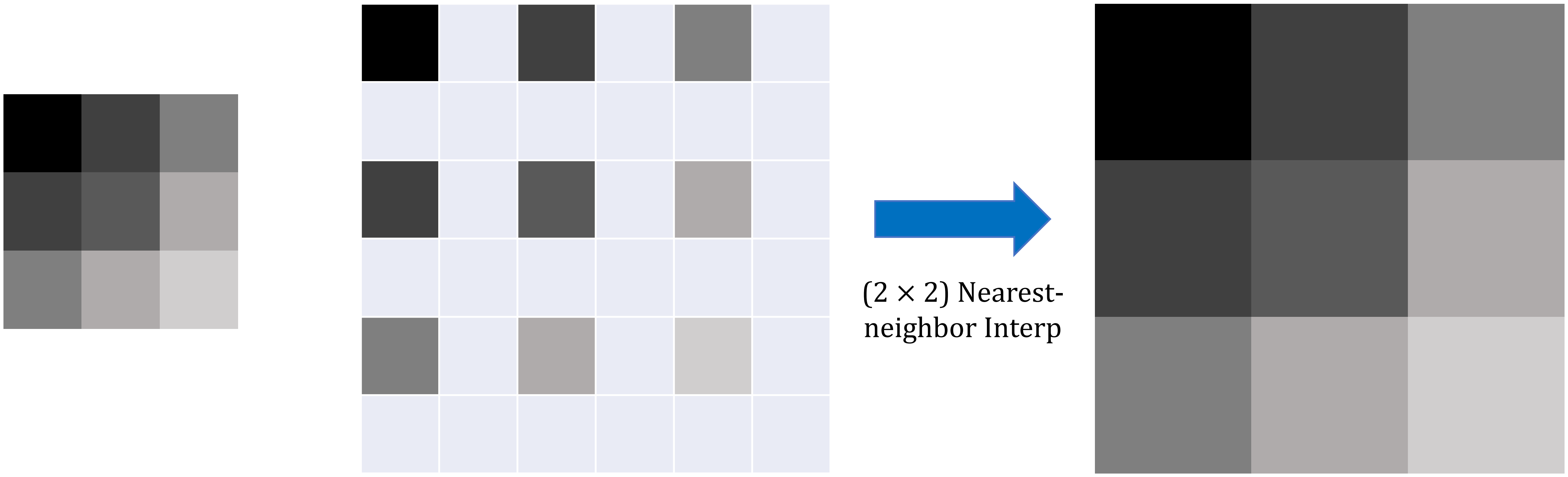
(2×2) Nearest-neighbor Interp

0	0	64	64	127	127
0	0	64	64	127	127
64	64	89	89	172	172
64	64	89	89	172	172
127	127	172	172	175	175
127	127	172	172	175	175

Interpolation Techniques

1. Nearest-Neighbor Interpolation

The nearest-neighbor interpolation or point-sampling takes the $n \times n$ neighbors of a point/pixel to adapt (copy) its value.



Interpolation Techniques

2. Bilinear Interpolation

For vertical and horizontal interpolation, we just use simple linear interpolation.

We base the weights of the values based on the distance of one pixel to the next

N

P_1 Value: 32 Position (2,0)	p1 Value: ? Position (2.2,0)	p2 Value: ? Position (2.4,0)	p3 Value: ? Position (2.6,0)	p4 Value: ? Position (2.8,0)	P_2 Value: 198 Position (3,0)
P_1 Value: 32 Position (2,0)	p1 Value: 65 Position (2.2,0)	p2 Value: 98 Position (2.4,0)	p3 Value: 131 Position (2.6,0)	p4 Value: 164 Position (2.8,0)	P_2 Value: 198 Position (3,0)

$$f(p) = \left(1 - \frac{1}{N-1}\right) f(P_1) + \left(\frac{1}{N-1}\right) f(P_2)$$

Interpolation Techniques

2. Bilinear Interpolation

For diagonal interpolation, we now consider the values of neighboring x and y pixels

$$f(p) = \frac{(x_1 - p_x)(y_2 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_1 + \frac{(p_x - x_1)(y_2 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_2 + \frac{(x_1 - p_x)(p_y - y_1)}{(x_2 - x_1)(y_2 - y_1)} P_3 + \frac{(p_x - x_1)(y_1 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_4$$

P_1 Value: 32 Position (2,0)	p1 Value: ? Position (2.2,0)	P_2 Value: 198 Position (3,0)
p2 Value: ? Position (2.4,0)	p3 Value: ? Position (2.6,0)	p4 Value: ? Position (2.8,0)
P_3 Value: 128 Position (2,1)	p4 Value: ? Position (2.8,0)	P_4 Value: 172 Position (3,1)

$x_1 = 2$
$x_2 = 3$
$y_1 = 0$
$y_2 = 1$

Interpolation Techniques

2. Bilinear Interpolation

For diagonal interpolation, we now consider the values of neighboring x and y pixels

$$f(p) = \frac{(x_1 - p_x)(y_2 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_1 + \frac{(p_x - x_1)(y_2 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_2 + \frac{(x_1 - p_x)(p_y - y_1)}{(x_2 - x_1)(y_2 - y_1)} P_3 + \frac{(p_x - x_1)(y_1 - p_y)}{(x_2 - x_1)(y_2 - y_1)} P_4$$

P_1 Value: 32 Position (2,0)	p1 Value: 114 Position (2.2,0)	P_2 Value: 198 Position (3,0)
p2 Value: 79 Position (2.4,0)	p3 Value: 132 Position (2.6,0)	p4 Value: 185 Position (2.8,0)
P_3 Value: 128 Position (2,1)	p4 Value: 150 Position (2.8,0)	P_4 Value: 172 Position (3,1)

$x_1 = 2$
$x_2 = 3$
$y_1 = 0$
$y_2 = 1$

Interpolation Techniques

3. Bicubic Interpolation

Bicubic interpolation is significantly slower compared to nearest neighbor and bilinear interpolation but produces less artifacts compared to the latter techniques. The interpolation algorithm per pixel can be expressed as:

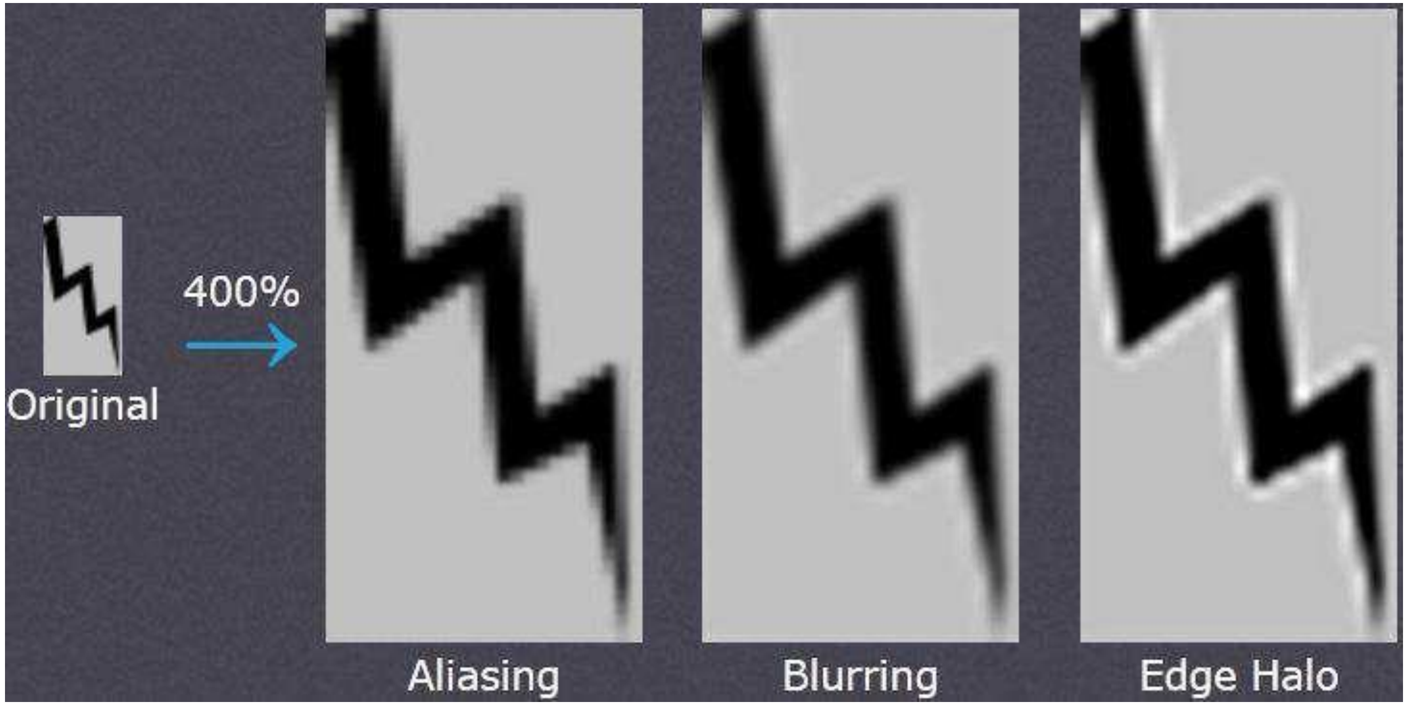
$$p(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j$$

Where x and y are the coordinates of the pixels and p is the color value the coefficient a_{ij} corresponds to the interpolation coefficient of the cubic interpolation.

Resampling Artifacts

Noise from zooming stems from sampling errors. Spatial expansion of images would leave spaces in between

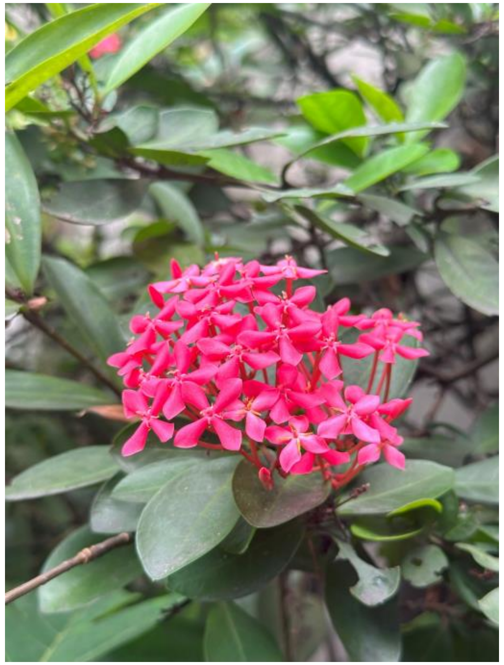
- Aliasing
- Blurring
- Halo



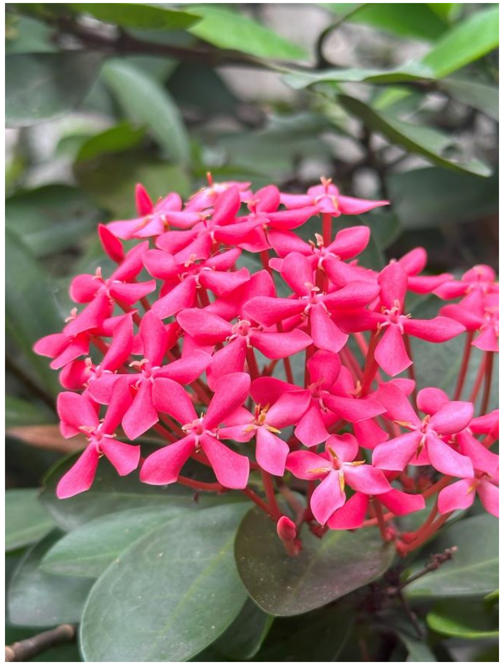
Zooming Noises

Normal Magnification

100% Magnification
(Optical Zoom)



200% Magnification
(Optical Zoom)



Taken using iPhone 14 Pro

Zooming Noises

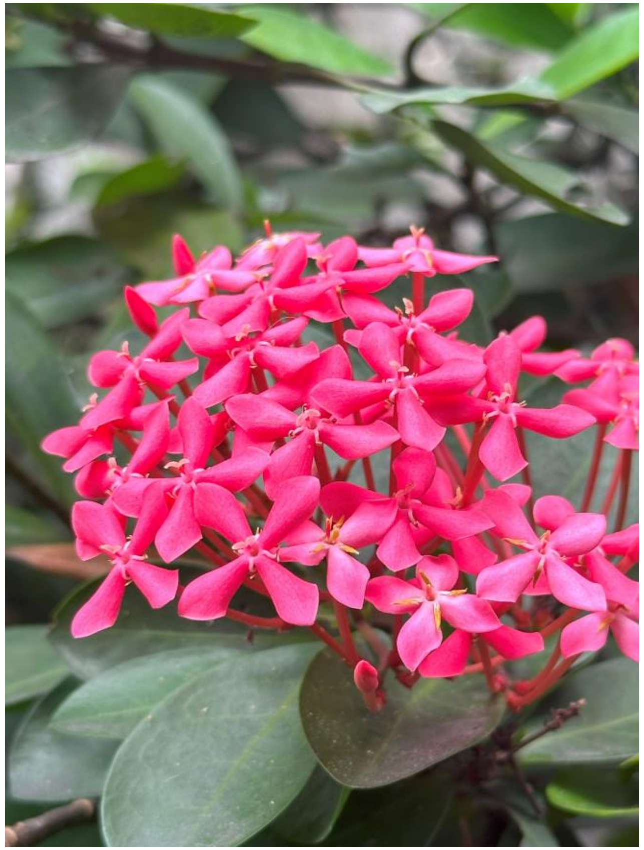
Normal Magnification

100% Magnification
(Geometric Zoom)

200% Magnification
(Geometric Zoom)

Taken using iPhone 14 Pro

Spot the Difference



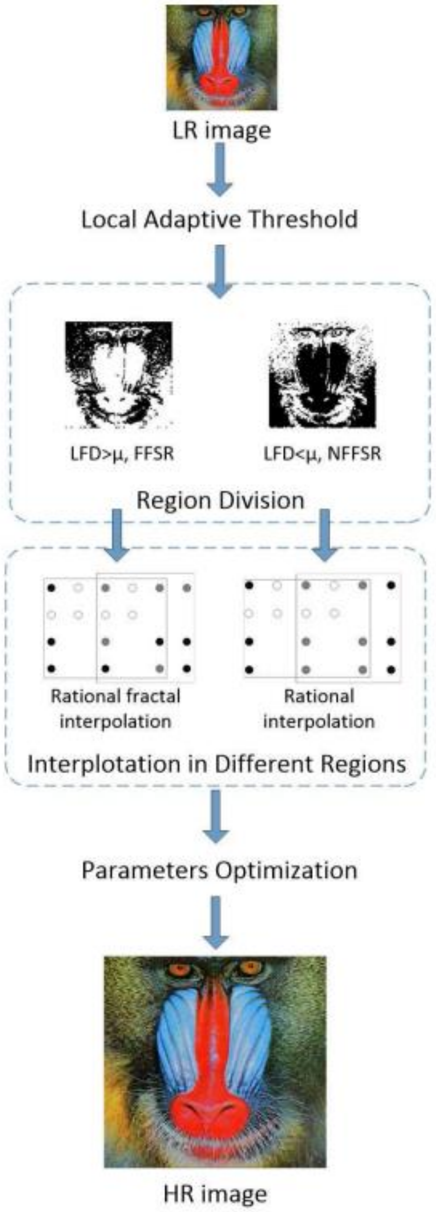
Optical Zoom

Geometric Zoom

Spot all the artifacts!

Advanced Solutions

- Adaptive Interpolation. Applying different interpolation techniques depending on region contents.
- Filters. Filters can remove artifacts using either high-pass or low-pass filtering



Thank you