The 27th IEEE International Conference on Image Processing 2020
Session: COM-03 – Image & Video Processing for Watermarking and Security

Information Hiding using Image Enhancement

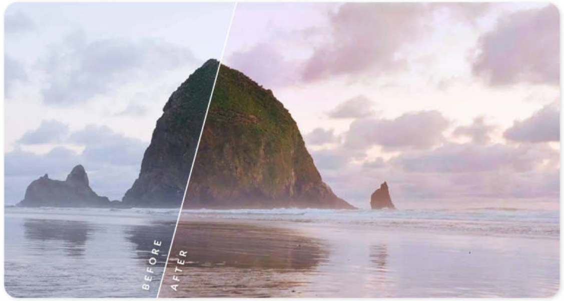
SIMYING ONG[1] AND KOKSHEIK WONG[2]

1 simying.ong@um.edu.my
2 wong.koksheik@monash.edu

1 University of Malaya, Malaysia
2 Monash University Malaysia

Image Enhancement

Definition: improves the quality of image by means of emphasizing the desired details or removing / suppressing the undesired noise / irrelevant information



Problem: What if we want to share additional data / add metadata into such images?

Information Hiding (IH)

Conventional IH Methods

- Least Significant Bit Insertion
- Histogram Shifting
- Difference Expansion
- Others...

Problem: These methods are NOT WIDELY ADAPTED into the usual operations performed by the users (extra steps needed) and they often caused image quality degradation.

What if we design an information hiding method as part of the image enhancement process?

Median Filtering - Revisit

Commonly utilized for noise removal.

67	32	31
45	**21**	22
83	42	46

Target pixel: {21}
$\omega = 3$

{67, 32, 31, 45, 21, 22, 83, 42, 46}

Collect all the pixels within neighbourhood in a linear form.

{21, 22, 31, 32, 42, 45, 46, 67, 83}

Sort the pixels in ascending order.

{21, 22, 31, 32, **42**, 45, 46, 67, 83}

	42	

Select the medium value and replace it with the target pixel.

Data Embedding

67	32	31
45	**21**	22
83	42	46

Target pixel: {21}
$\omega = 3$

$\{67, 32, 31, 45, 21, 22, 83, 42, 46\}$

Collect all the pixels within neighbourhood in a linear form.

$\{21, 22, 31, 32, 42, 45, 46, 67, 83\}$

Sort the pixels in ascending order.

(reusing previous example)

Notation:

r = number of partitions

Γ_i = i-th partition where $i = \{0, \dots, r\}$

$r = 2$

If secret is ‘0’ $\longrightarrow$ $\Gamma_0 = \{21, 22, 31, 32\}$

If secret is ‘1’ $\longrightarrow$ $\Gamma_1 = \{42, 45, 46, 67, 83\}$

Say the secret is ‘1’

	46	

Data Extraction

Same neighbourhood is considered

	46	

BUT the challenging part is that the **neighbourhood might have different values** → because they all are median-filtered (plus hidden info.)

1. Gather the pixels, linearized, sorted them.

2. If $r = 2$, the median value Υ is considered.

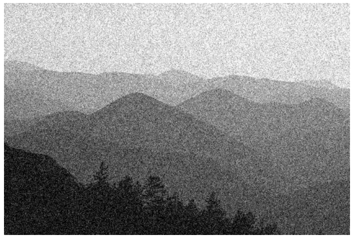
3. If $X' < \Upsilon$, secret '0' is extracted. Else, '1' is extracted.

Experiment Settings

- Prototype built using MATLAB 2018b.
- Test Images: BSD300 dataset – 300 test images (Grayscale)
  (481x321 pixels or 321x481 pixels)




- Noise generation: *imnoise* function in Matlab – Gaussian and Speckle

Gaussian Noise 0.01 -> 0.05


Speckle Noise 0.01 -> 0.05

- Parameter settings: $\omega = 3, 5, 7$ and $r = 2$

Experiment results
Embedding Capacity

- Number of bits that can be embedded into the image.

$$\mathrm{E} = \lfloor \log r \rfloor \times \big((M - \omega + 1) \times (N - \omega + 1) \big)$$

where M and N are the size of the image.

ω	3	5	7
$r = 2$	~ 0.99 bpp	~ 0.98 bpp	~ 0.97 bpp

- Increasing r can increase the embedding capacity.

Experiment results
Image Quality

Table 1: Average PSNR [db] for output images with **Speckle Noise.**

w	γ_i	Noise Addition				
		0.01	0.02	0.03	0.04	0.05
Noise Image		26.3	23.4	21.8	20.6	19.7
3	γ_5	27.3	26.0	25.1	24.3	23.7
	$\{\gamma_1, \gamma_9\}$	18.2	16.8	15.9	15.2	14.6
	$\{\gamma_2, \gamma_8\}$	20.9	19.6	18.6	17.9	17.3
	$\{\gamma_3, \gamma_7\}$	23.3	22.1	21.2	20.5	19.9
	$\{\gamma_4, \gamma_6\}$	26.2	24.8	23.9	23.1	22.5
5	γ_{25}	25.5	24.9	24.4	24.0	23.7
	$\{\gamma_1, \gamma_{25}\}$	15.4	14.3	13.5	13.0	12.5
	$\{\gamma_4, \gamma_{21}\}$	19.3	18.4	17.6	17.0	16.5
	$\{\gamma_7, \gamma_{18}\}$	22.0	21.2	20.5	19.9	19.4
	$\{\gamma_{10}, \gamma_{15}\}$	24.4	23.7	23.2	22.7	22.3
7	γ_{25}	24.4	24.1	23.7	23.5	23.2
	$\{\gamma_1, \gamma_{49}\}$	14.0	13.0	12.4	11.9	11.4
	$\{\gamma_6, \gamma_{44}\}$	17.5	16.6	16.0	15.4	14.9
	$\{\gamma_{11}, \gamma_{39}\}$	19.8	19.1	18.4	17.9	17.4
	$\{\gamma_{16}, \gamma_{34}\}$	22.0	21.4	20.8	20.4	20.0

When ω increases, median filtering enhancement effects are weaker (applicable even when our proposed method is not applied).

Blue → Original (Median Filter)

Red → `0'
Green → '1'

Experiment results

Image Quality

When ω is small and when selected pixels for replacement is near median value, our proposed method can still enhance the image AND embed data.

Table 2: Average SSIM for output images with **Gaussian Noise.**

w	γ_i	Noise Addition				
		0.01	0.02	0.03	0.04	0.05
Noise Image		0.355	0.354	0.354	0.353	0.352
3	γ_5	0.556	0.554	0.552	0.550	0.547
	$\{\gamma_1, \gamma_9\}$	0.108	0.108	0.108	0.108	0.108
	$\{\gamma_2, \gamma_8\}$	0.201	0.201	0.201	0.201	0.201
	$\{\gamma_3, \gamma_7\}$	0.322	0.321	0.321	0.320	0.319
	$\{\gamma_4, \gamma_6\}$	0.470	0.469	0.468	0.466	0.464
5	γ_{13}	0.574	0.571	0.569	0.566	0.563
	$\{\gamma_1, \gamma_{25}\}$	0.044	0.044	0.044	0.045	0.045
	$\{\gamma_4, \gamma_{21}\}$	0.135	0.135	0.135	0.135	0.135
	$\{\gamma_7, \gamma_{18}\}$	0.258	0.257	0.257	0.256	0.256
	$\{\gamma_{10}, \gamma_{15}\}$	0.452	0.451	0.450	0.447	0.446
7	γ_{25}	0.559	0.557	0.554	0.551	0.547
	$\{\gamma_1, \gamma_{49}\}$	0.029	0.029	0.029	0.029	0.029
	$\{\gamma_6, \gamma_{44}\}$	0.080	0.080	0.080	0.080	0.080
	$\{\gamma_{11}, \gamma_{39}\}$	0.148	0.148	0.148	0.148	0.149
	$\{\gamma_{16}, \gamma_{34}\}$	0.265	0.264	0.264	0.263	0.262

Experiment results

Image Quality

Table 1: Average PSNR [db] for output images with **Speckle Noise.**

w	γ_i	Noise Addition				
		0.01	0.02	0.03	0.04	0.05
Noise Image		26.3	23.4	21.8	20.6	19.7
3	γ_5	27.3	26.0	25.1	24.3	23.7
	$\{\gamma_1, \gamma_9\}$	18.2	16.8	15.9	15.2	14.6
	$\{\gamma_2, \gamma_8\}$	20.9	19.6	18.6	17.9	17.3
	$\{\gamma_3, \gamma_7\}$	23.3	22.1	21.2	20.5	19.9
	$\{\gamma_4, \gamma_6\}$	26.2	24.8	23.9	23.1	22.5
5	γ_{25}	25.5	24.9	24.4	24.0	23.7
	$\{\gamma_1, \gamma_{25}\}$	15.4	14.3	13.5	13.0	12.5
	$\{\gamma_4, \gamma_{21}\}$	19.3	18.4	17.6	17.0	16.5
	$\{\gamma_7, \gamma_{18}\}$	22.0	21.2	20.5	19.9	19.4
	$\{\gamma_{10}, \gamma_{15}\}$	24.4	23.7	23.2	22.7	22.3
7	γ_{25}	24.4	24.1	23.7	23.5	23.2
	$\{\gamma_1, \gamma_{49}\}$	14.0	13.0	12.4	11.9	11.4
	$\{\gamma_6, \gamma_{44}\}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	$\{\gamma_{11}, \gamma_{39}\}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
	$\{\gamma_{16}, \gamma_{34}\}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Table 2: Average SSIM for output images with **Gaussian Noise.**

w	γ_i	Noise Addition				
		0.01	0.02	0.03	0.04	0.05
Noise Image		0.355	0.354	0.354	0.353	0.352
3	γ_5	0.556	0.554	0.552	0.550	0.547
	$\{\gamma_1, \gamma_9\}$	0.108	0.108	0.108	0.108	0.108
	$\{\gamma_2, \gamma_8\}$	0.201	0.201	0.201	0.201	0.201
	$\{\gamma_3, \gamma_7\}$	0.322	0.321	0.321	0.320	0.319
	$\{\gamma_4, \gamma_6\}$	0.470	0.469	0.468	0.466	0.464
5	γ_{13}	0.574	0.571	0.569	0.566	0.563
	$\{\gamma_1, \gamma_{25}\}$	0.044	0.044	0.044	0.045	0.045
	$\{\gamma_4, \gamma_{21}\}$	0.135	0.135	0.135	0.135	0.135
	$\{\gamma_7, \gamma_{18}\}$	0.258	0.257	0.257	0.256	0.256
	$\{\gamma_{10}, \gamma_{15}\}$	0.452	0.451	0.450	0.447	0.446
7	γ_{25}	0.559	0.557	0.554	0.551	0.547
	$\{\gamma_1, \gamma_{49}\}$	0.029	0.029	0.029	0.029	0.029
	[illegible]	[illegible]	[illegible]	0.080	0.080	0.080
	[illegible]	[illegible]	[illegible]	0.148	0.148	0.149
	[illegible]	[illegible]	[illegible]	0.264	0.263	0.262

While increasing the noise level, average PSNR decreases BUT average SSIM are maintained at consistent level for various noise level.

Experiment results

Image Quality

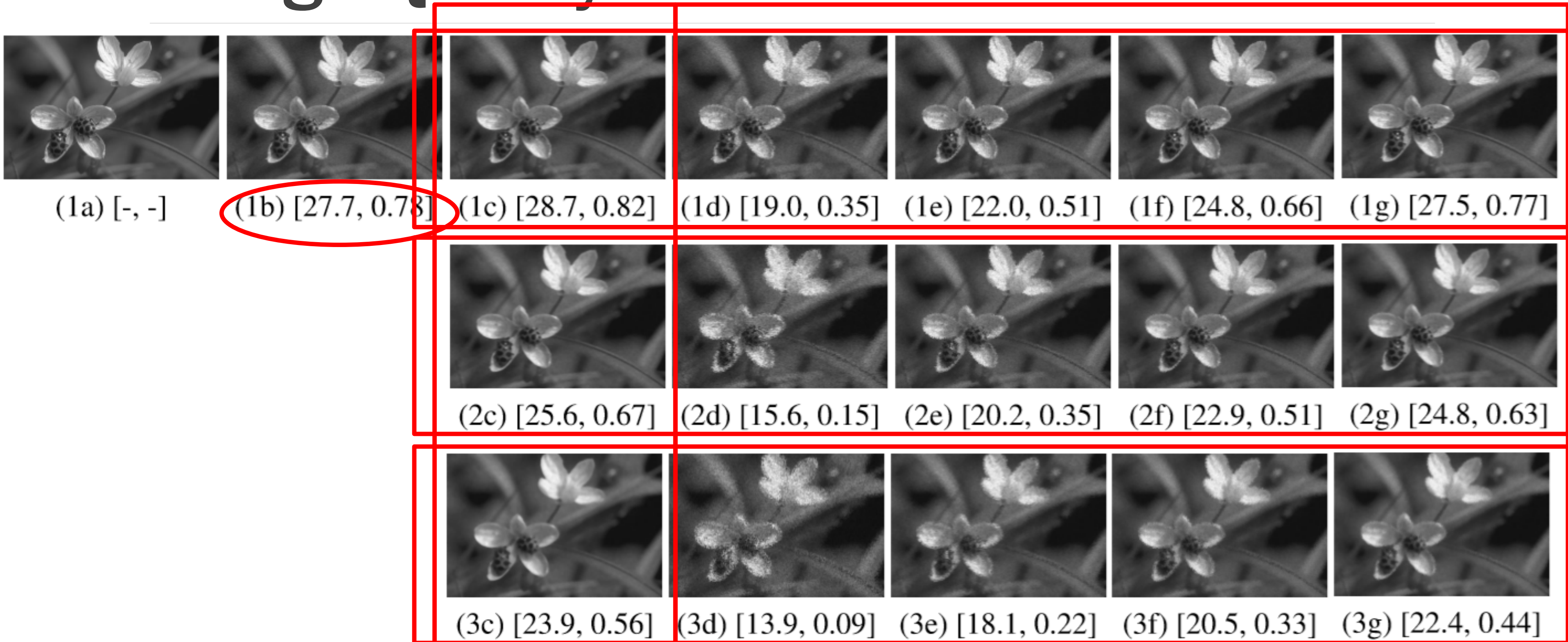


Fig. 2. Output images with Speckle noise (level = 0.01) added and their respective [PSNR, SSIM] values. Row 1 to Row 3 correspond to the output when $w = \{3, 5, 7\}$. Sub-figures 1(a) and 1(b) are the reference and generated-noise images, respectively. The third column shows Median-filtered image (without data embedding) for various w. Similarly, the forth to seventh columns show Median-filtered-embedded images with the pixel pair settings recorded in **Table 1**.

Experiment results

Data Extraction Error Rate

During data extraction, the neighbourhood might have different values → they are median-filtered.

→ **Data extraction error might occurred.**

Table 3: Average data extraction error rate [%] for images with Speckle noise.

w	γ_i	Noise Addition				
		0.01	0.02	0.03	0.04	0.05
3	$\{\gamma_1, \gamma_9\}$	15.7	15.9	15.9	16.0	16.0
	$\{\gamma_2, \gamma_8\}$	13.6	13.7	13.7	13.7	13.7
	$\{\gamma_3, \gamma_7\}$	14.2	13.9	13.8	13.8	13.7
	$\{\gamma_4, \gamma_6\}$	24.4	23.3	22.8	22.4	22.1
5	$\{\gamma_1, \gamma_{25}\}$	8.8	8.9	8.9	8.9	8.9
	$\{\gamma_4, \gamma_{21}\}$	6.2	6.2	6.2	6.2	6.2
	$\{\gamma_7, \gamma_{18}\}$	6.3	6.1	6.0	6.0	6.0
	$\{\gamma_{10}, \gamma_{15}\}$	13.7	12.4	11.7	11.3	11.0
7	$\{\gamma_1, \gamma_{49}\}$	6.1	6.1	6.2	6.2	6.2
	$\{\gamma_6, \gamma_{44}\}$	3.8	3.8	3.7	3.7	3.7
	$\{\gamma_{11}, \gamma_{39}\}$	3.7	3.6	3.5	3.5	3.5
	$\{\gamma_{16}, \gamma_{34}\}$	4.2	3.9	3.8	3.7	3.6

Error rates are ranged within 24.4% to 3.5%

24.4% is when ω is small and the selected pixels for data embedding are near (i.e., 4th pixel to represent ‘0’ and 6th pixel to represent ‘1’).

3.5% is when ω is large and the selected pixels for data embedding are far (i.e., 11th pixel to represent ‘0’ and 39th pixel to represent ‘1’).

Easier to distinguish the embedded bit.

Discussion

To improve data extraction efficiency

- Same copies of data can be embedded for 3 times – majority vote is used to decide the secret.
- Extra postprocessing step after data embedding to ensure correct data extraction.
- Use error correction code (e.g., Hamming code).

To improve robustness against unauthorized data extraction

- Use secret key to randomize the parameter ω, r and the selected pixels for data embedding in each partition.

Conclusion and Future Work

- Information is embedded while executing image enhancement step.
- As a proof of concept, Median Filtering is redesigned to enable information hiding.
- Under various settings, the proposed method can embed up to 0.99 bpp, with the data extraction error rate ranging from 3.5% to 24.4%.

Future work

consider colour image, and investigate the correlation between various colour channels for deploying filtering-embedding framework.

Q/A

Feel free to contact us at:

simying.ong@um.edu.my

wong.koksheik@monash.edu