ENSE 496AE
OpenStego for Watermarking

•••

Jeremy Cross 200319513
Taylen Jones 200354271
Bipin Suram 200439429
February 4 2020

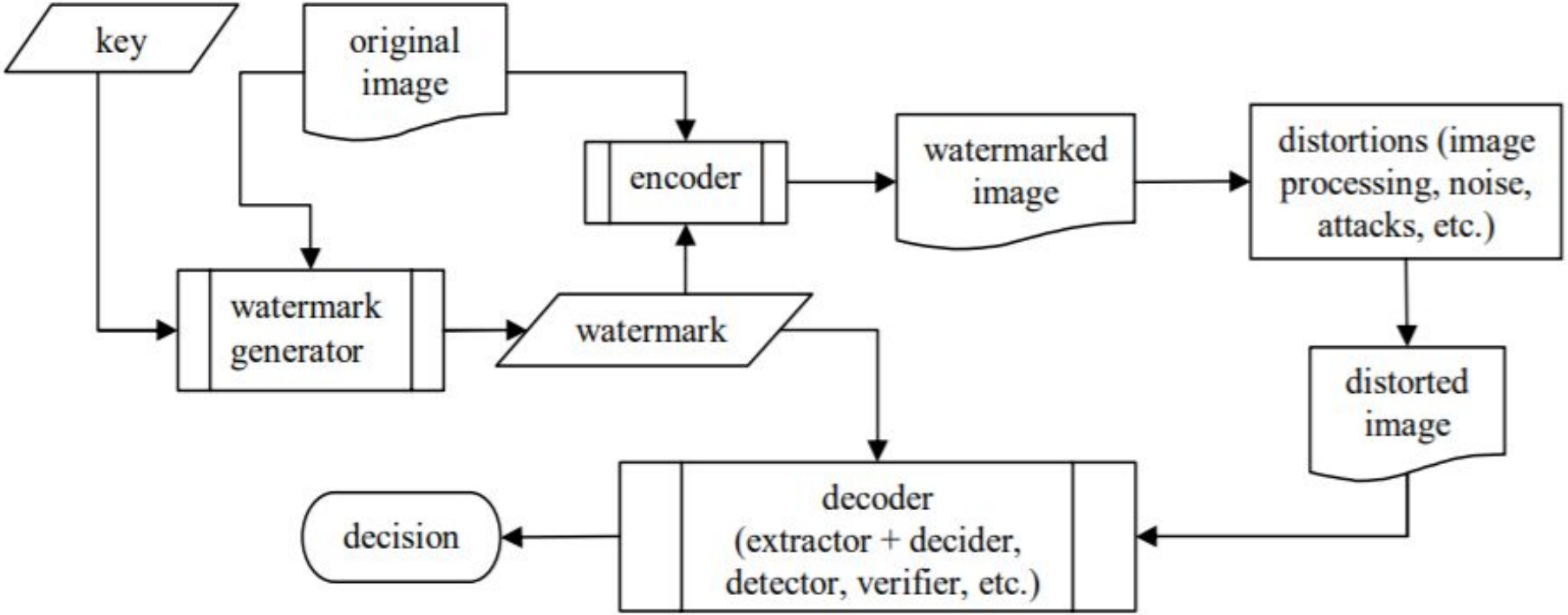
Why use Watermarking?

- The process of embedding a covert marker that identifies ownership of the cover image
- Can be used to verify the authenticity or integrity of the cover image and identify its owner
- Digital watermarks should only be perceptible under certain conditions using some algorithm and imperceptible any other time
- If the watermark in the cover image is perceivable then it is not very useful (the idea is to keep it hidden)

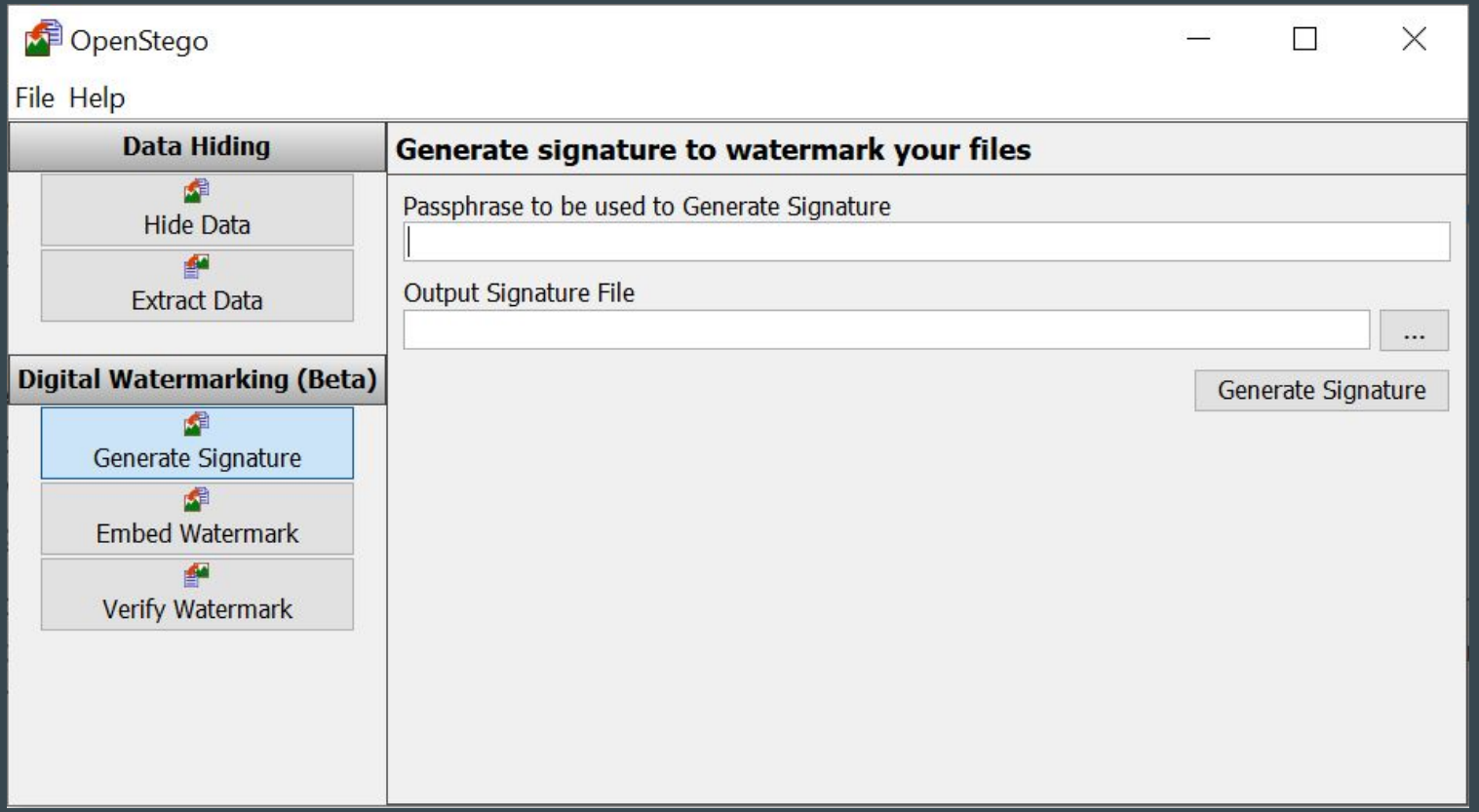
OpenStego

- Offers both data hiding and watermarking (we are just looking at watermarking)
- OpenStego is purely written in java
- Supports password based encryption for an additional layer of security (only offered for data hiding)
- Utilizes plugin based architecture
- The two plugins currently supported are RandomLSB for data hiding and Dugads algorithm for watermarking
- new plugins can be created for other algorithms

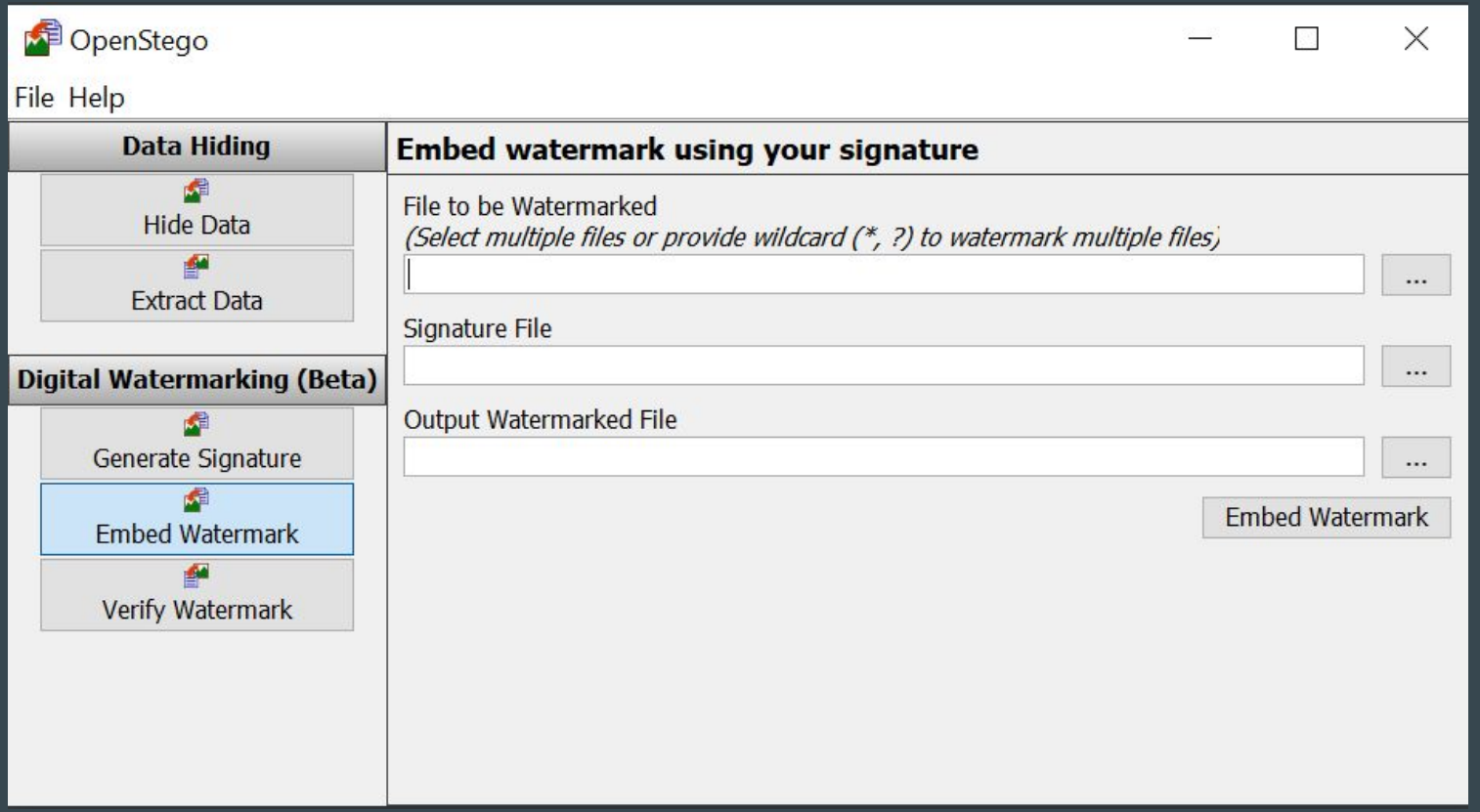
General scenario



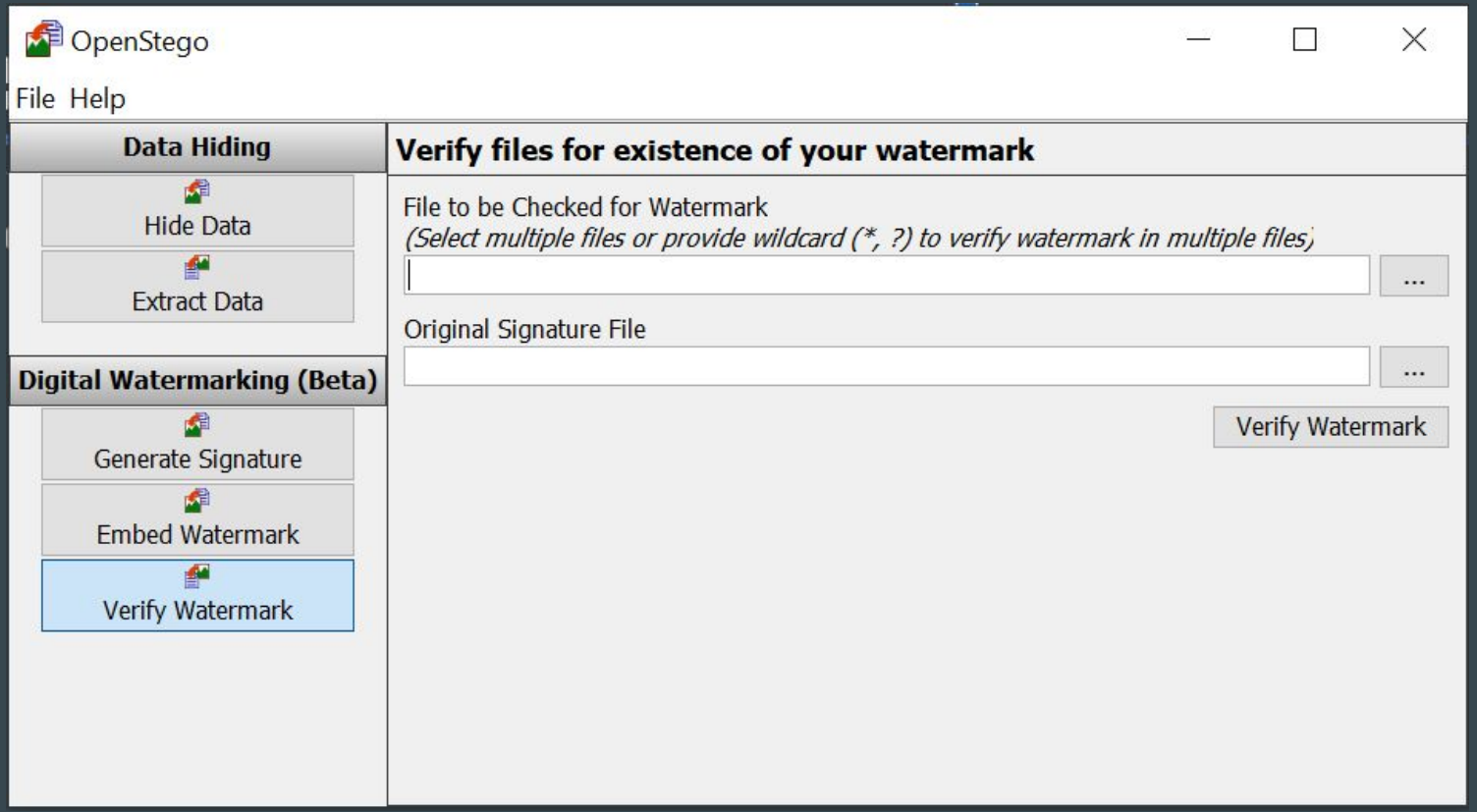
Generating Signature



Embed Watermark



Verify Watermark



Results with Watermarking Images

- PNG and BMP file types largely increase in file size
- JPG files have the opposite effect where they largely decrease in file size
- Cropping any single edge can make the strength to zero
- Modifying the file name does not change strength
- Rotating the image does not change strength
- Re-locating the file does not change the strength
- The strength changes when changing the brightness and applying any filters
- The compression on the marked file does not change the strength
- Size of the output file does not depend on signature used

Size of the file

- Actual image (jpg) - 1331 KB

- Image size after watermarking -

  When saved in jpg - 606 KB (88% strength)

  When saved in png - 7259 KB (100% strength)

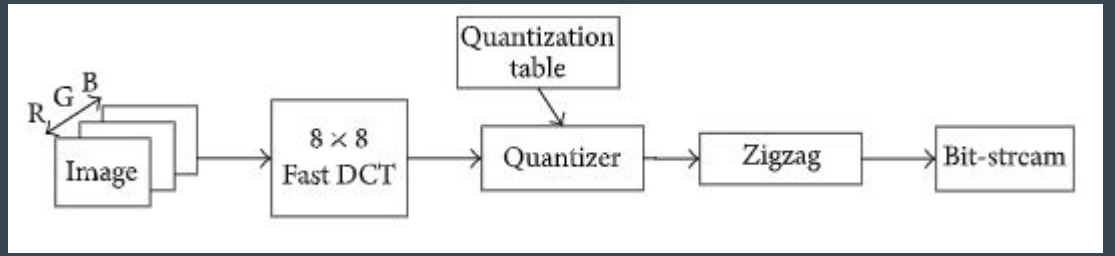
  When saved in bmp - 11057 KB (100% strength)

Effects of Compression

The strength of watermarking is not changed by compressing the output file.

	Marked file	Compressed file
Size -	606 KB	362 KB
Strength -	88%	88%

Type of Insertion



Insertion technique that can be used with watermarking:

- DCT (Discrete Cosine Transformation)
- Image is divided into 8x8 pixel blocks
- This allows for 64 (0 - 63) frequency coefficients
- Low frequency coefficients represent lighter color regions
- High frequency coefficients represent darker color regions
- The high frequency coefficients don't contribute very much to the overall image
- Removing the high frequency data is known as quantization
- Divide the coefficients by quantization table
- Compress the data further by creating a bitstream comprised of the remaining values using zigzag pattern

References

- https://www.openstego.com/index.html
- https://pdfs.semanticscholar.org/a9c0/9195058d0bc65f91fc361a13a5de2a870363.pdf
- https://www.youtube.com/watch?v=Q2aEzeMDHMA
- https://www.hindawi.com/journals/mpe/2017/7401845/

Questions?

Post Presentation Questions

- Compare a watermarked image with a screen capture of the watermarked image and observe the results

  When comparing the images of the watermarked image and the screen capture of watermarked image the strength of the screen captured watermarked image was 0% which confirms that the watermark is not present within the screen capture. This can be seen with an example on the next slide.

- Crop a watermarked image and observe the results

  Cropping any part of the watermarked image affects the strength.
  The strength becomes 0% if any single edge is cropped.

Comparing watermark image with screen capture of watermarked Image

Strength: 88%

Strength: 0%