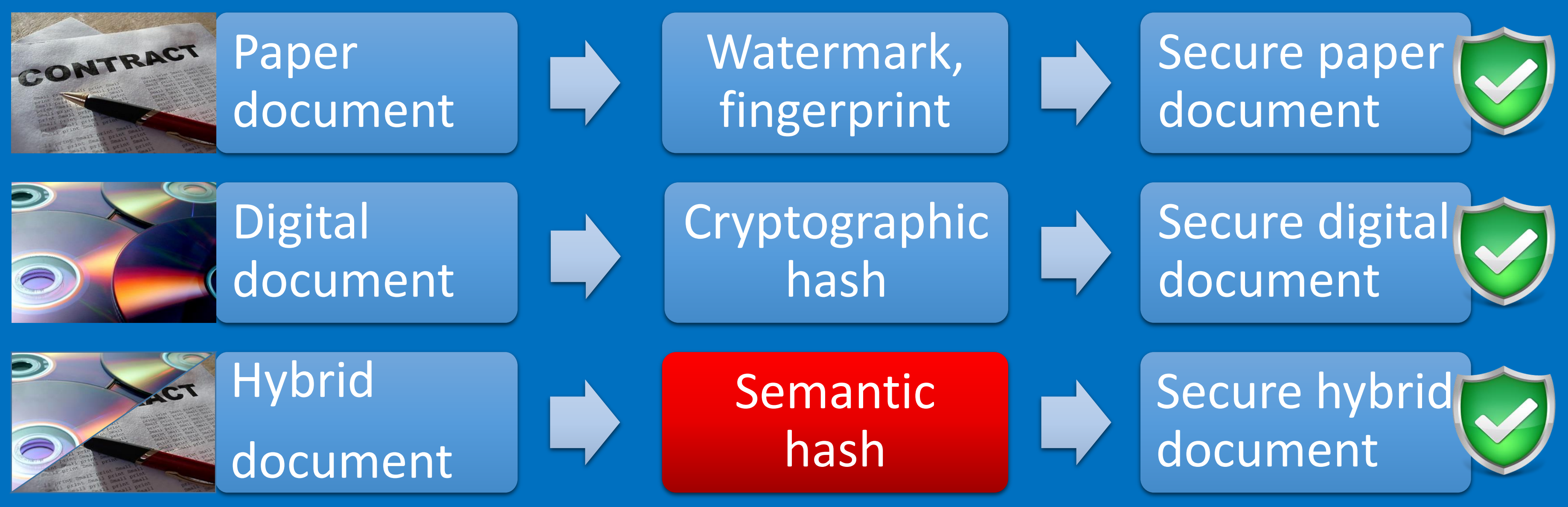
Document semantic hashing for hybrid security

Sébastien Eskenazi, Petra Gomez-Krämer, Jean-Marc Ogier - Laboratoire L3i - La Rochelle, France

Context: document security

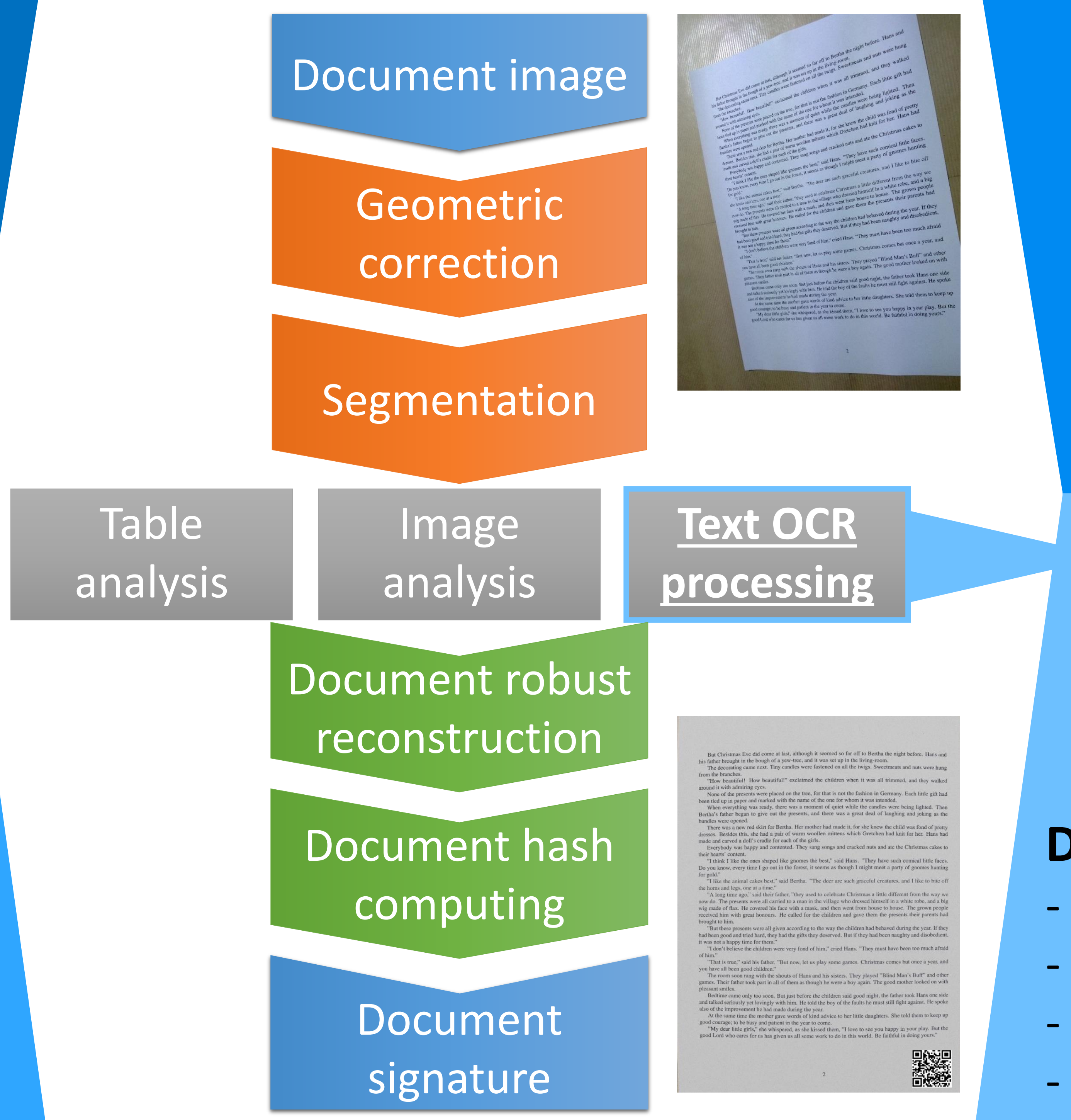


Paper document → Watermark, fingerprint → Secure paper document

Digital document → Cryptographic hash → Secure digital document

Hybrid document → Semantic hash → Secure hybrid document

Hash computation process



Document image

Geometric correction

Segmentation

Table analysis

Image analysis

Text OCR processing

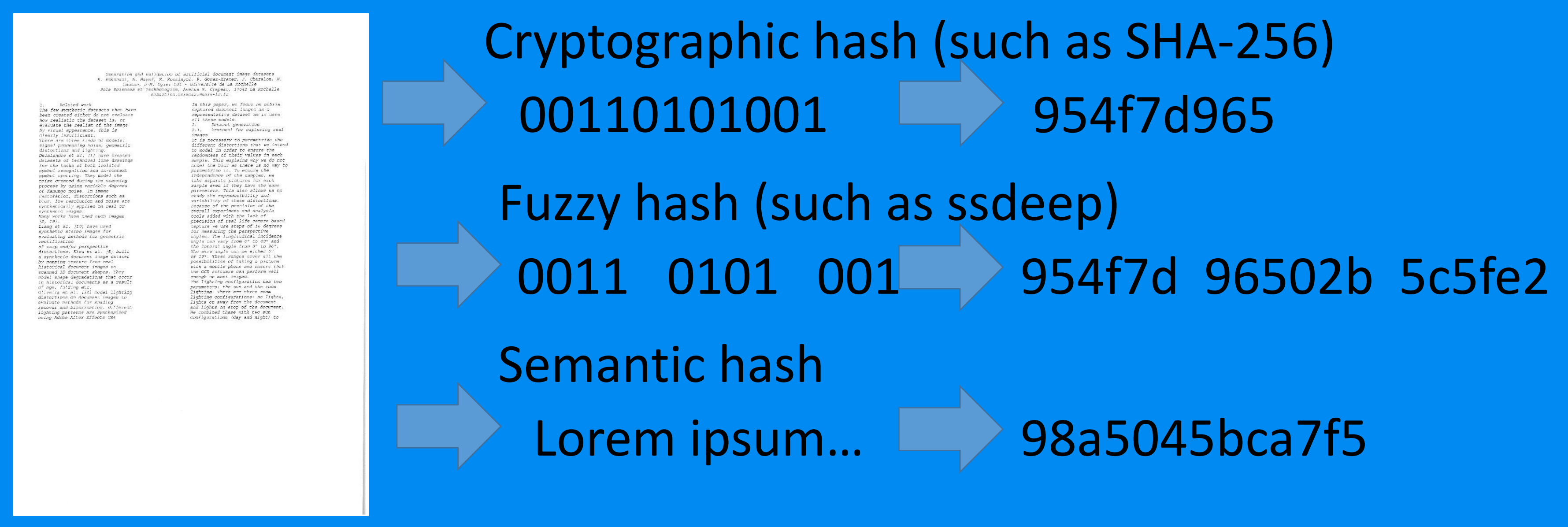
Document robust reconstruction

Document hash computing

Document signature

954f7d96502b5c5fe2e98a5045bca7f5

What is a semantic hash ?



Cryptographic hash (such as SHA-256)

00110101001 → 954f7d965

Fuzzy hash (such as ssdeep)

0011 0101 001 → 954f7d 96502b 5c5fe2

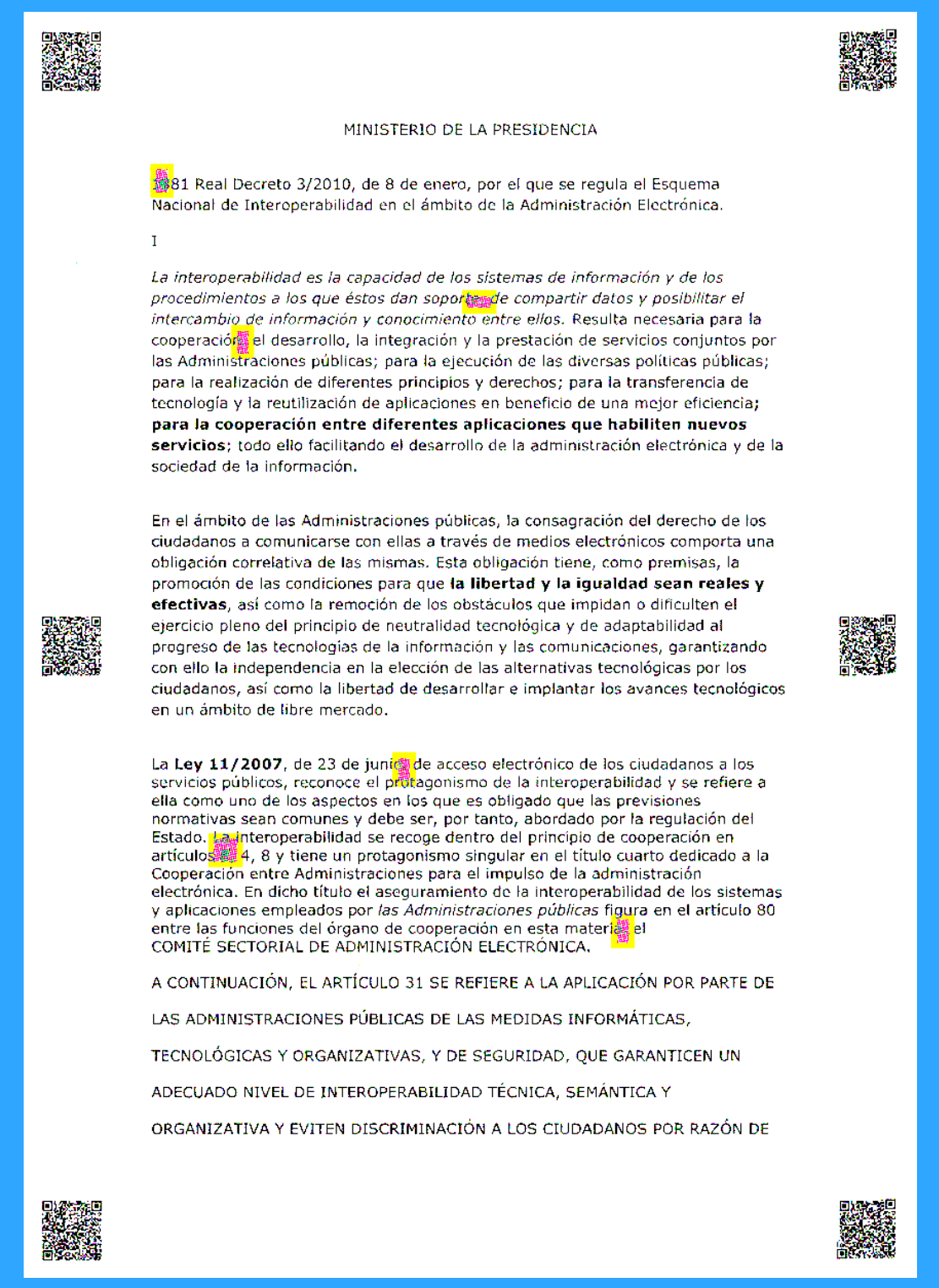
Semantic hash

Lorem ipsum… → 98a5045bca7f5

State of the art and remaining challenges

SIGNED project [1]

- Cut the document image in squares of 64 by 64 pixels
- Apply a Haar Discrete Wavelet Transform on each square
- Create a fuzzy hash
- Other undisclosed pre- and post-processing steps.



Document secured by the SIGNED project, it takes 6 2D-barcodes to embed the signature

Strengths	Weaknesses
Probability of false alarm <0.001	Cannot detect the replacement of dots by commas with an error <0.001
Probability of missed detection <0.001 for the replacement of digits	Cannot detect a manipulation smaller than 64x64 pixels at 600dpi (42x42 required)
Collision probability <0.001	Throughput >5s per page
Compatible with current scanners and printers	Hash size >4kB (between 4.8 and 170 kB)

References:

[1]A. Malvido Garcia: Secure Imprint Generated for Paper Documents (SIGNED). Technical Report December 2010, Bit Oceans (2013)

[2] S. Eskenazi, P. Gomez-Krämer, J.-M. Ogier: When document security brings new challenges to document analysis. International Workshop on Computational Forensics (IWCF), (2014)

For further information

l3i.univ-larochelle.fr

Conclusion: We need to study and improve the robustness of document analysis algorithms.

Best case scenario results:

- Accuracy : 99.83%
- Probability of false positive: 53% !!!
- Collision probability : 0.2%
- Other criteria are OK (throughput, hash size…)

False positives:

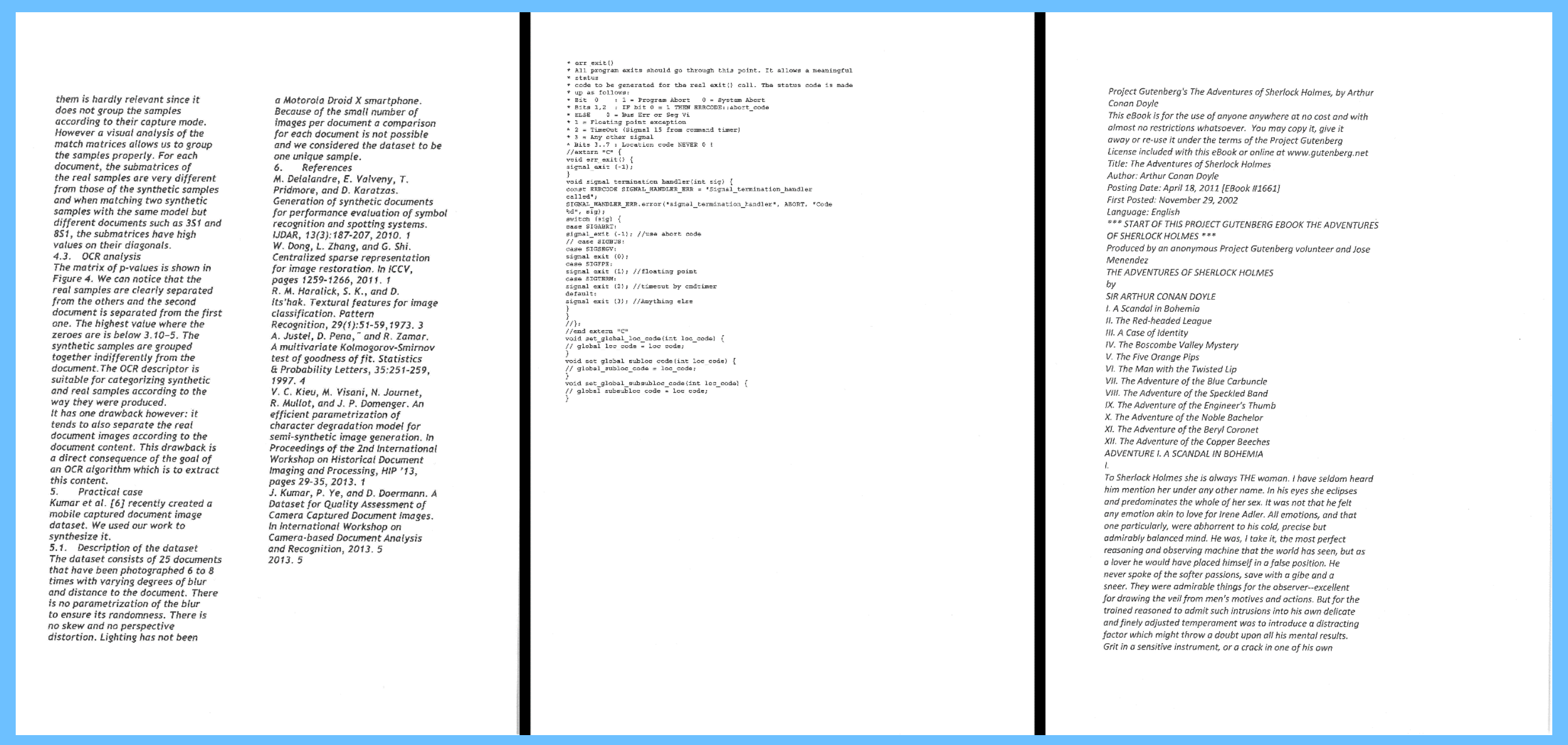
- Occurs when two identical documents have different hashes
- Related to the robustness of the OCR algorithm
- Computed as a random draw of two copies of the same document:

$$P_i = \sum_{j=1}^{n_i}\left(\frac{n_{ij}}{\sum_{k=1}^{n_i} n_{ij}} \times \frac{n_{ij}-1}{\sum_{k=1}^{n_i} n_{ij}-1}\right)$$

- n documents
- n_i hashes per document
- Each hash is present n_{ij} times

Text OCR processing [2]

Test of Tesseract on 28512 document images



Dataset:

- 22 texts
- 6 fonts
- 3 font sizes
- 4 font emphasis
- 3 printers
- 3 scanners
- 3 scanning resolutions

Test protocol:

- Run Tesseract on the document image
- Post processing : alphabet reduction
- Compute the SHA-256 hash of the document

Alphabet reduction	
Character	**Replacement**
Empty line	Removed
Tabulation and space	Removed
— (long hyphen)	- (short hyphen)
´, ` (left and right apostrophes)	' (centered apostrophe)
‶,″,'' (left and right quotes, double apostrophe)	" (centered quote)
I, l, 1 (capital i, 12th letter of the alphabet, number 1)	| (vertical bar)
O (capital o)	0 (zero)
fi (ligature)	fi (two letters f and i)
fl (ligature)	fl (two letters f and l)