Academy of the Federal Guard Service

Estimation of Watermark Embedding Capacity with Line Space Shifting

Alexander Kozachok, Dr. Sci.
Sergey Kopylov

September, 25

Information leaks

Figure 1. The information leaks number 2004-2019 (in thousands)

The incidents involving the leakage share of paper text documents in 2019 was 13.4%.

Watermark based on line space shifting

Watermarking text document formats:

- RTF (Rich Text Format);
- DOC (Word .doc binary format);
- DOCX (Word Extension to the Office Open XML .docx file format);
- PDF (Portable Document Format);
- ODT (Open Document Text).

A multicomponent approach to watermark embedding:

– line spacing increase by the set values of Δ_λ between adjacent lines is interpreted as "1", "2", "3";

– line spacing decreasing by the set values Δ_λ is interpreted as "-1", "-2", "-3";

– unchanged line spacing value – as "0".

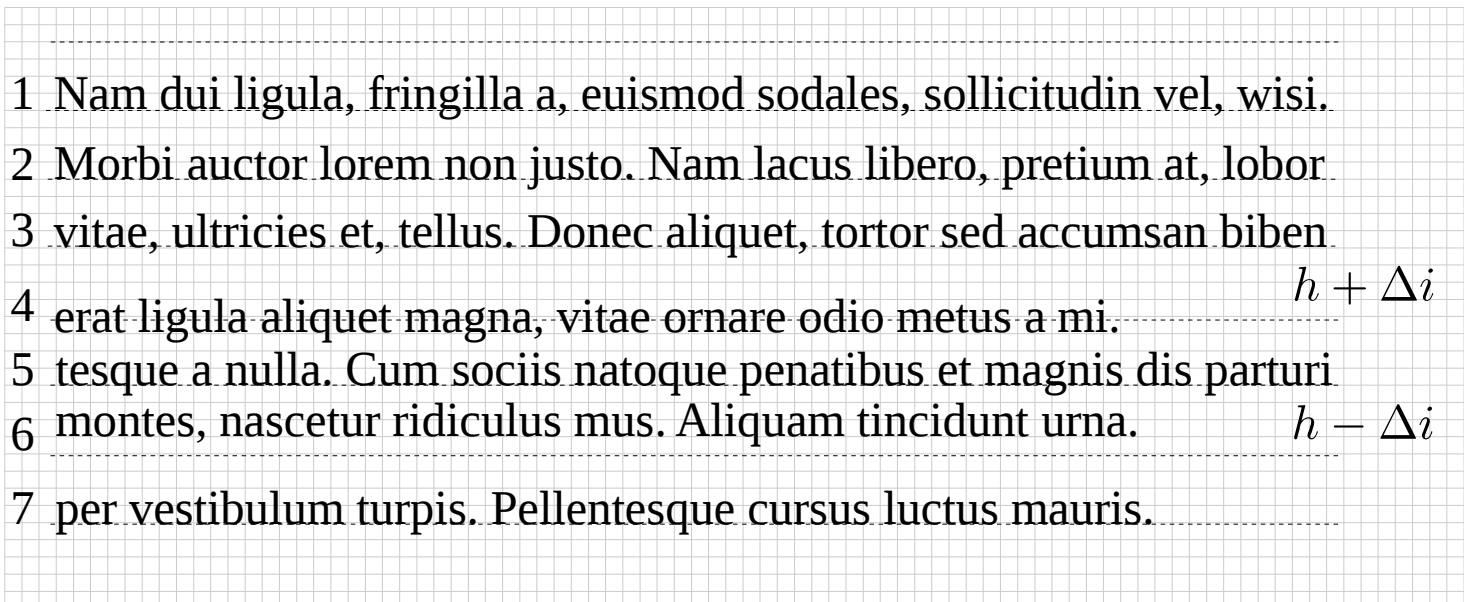

Figure 2. Line space shifting

Watermark embedding

Data: Text document TD_O, embed value Δ, embed information L

Result: Marked text document TD_S

1 $N \leftarrow$ **CountLines** (TD_O)

2 **if** $N \geqslant 4$ **then**

3 **if** $N == 4$ **then**

4 $\lambda \leftarrow 3$

5 $I \leftarrow$ **Coding** (L, λ)

6 $TD_O \leftarrow$ **Embed** $(TD_O, I, \lambda, \Delta)$

7 **else**

8 $\lambda \leftarrow$ **Input** *(L, N)*

9 $I \leftarrow$ **Coding** (L, λ)

10 $TD_O \leftarrow$ **Embed** $(TD_O, I, \lambda, \Delta)$

11 $TD_s \leftarrow TD_0$

12 **return** TD_s

Figure 3. Watermark embedding algorithm

1 **Function** `Embed`$(TD_O, \Delta, \lambda, I)$

2 $Len \leftarrow$ **GetLength** (I)

3 **if** $N > (Len + 1)$ **then**

4 **if** $\lambda \mod 2 \neq 0$ **then**

5 **for** $i \leftarrow 0$ **to** $(N - 2)$ **do**

6 $j \leftarrow i \bmod Len$

7 **if** $I_j < \dfrac{\lambda - 1}{2}$ **then**

8 $\Delta' \leftarrow I_j \cdot \Delta + \Delta$

9 $TD_O \leftarrow$ **Decrease** (TD_O, i, Δ')

10 **else**

11 **if** $I_j > \dfrac{\lambda - 1}{2}$ **then**

12 $\Delta' \leftarrow (\lambda \mod I_j) \cdot \Delta$

13 $TD_O \leftarrow$ **Increase** (TD_O, i, Δ')

14 **return** TD_O

Figure 4. Function Embed

Watermark extraction

Data: Image Im, embedding components value λ
Result: Embed information L

$Im_{proc} \leftarrow$ **ImageProcessing** (Im)
$Im_{text} \leftarrow$ **FindBlocks** (Im_{proc})
$sinogram \leftarrow$ **RadonTransform** (Im_{text})
$R \leftarrow$ **RMSCalc** $(sinogram)$
$rot \leftarrow$ **ArgMax** (R)
$row \leftarrow sinogram[rot]$
$M \leftarrow$ **CalculatePicks** (row)
$modes_o \leftarrow$ **CalculateModes** (M, λ)
$M_C \leftarrow$ **CorrectErrors** $(M, modes)$
$modes_c \leftarrow$ **CalculateModes** (M_C, λ)
if **Std** $(M_C) < 0{,}7$ **then**
    $m_i \leftarrow 0, i = \overline{1,|M_C|}$
    $C \leftarrow \{0\}$
else
    if $\lambda > 3$ **then**
        $C \leftarrow$ **KmeanClassification** $(M_C, \lambda, modes_c)$
    else
        $C \leftarrow$ **GMMClassification** $(M_C, \lambda, modes_c)$
$I' \leftarrow$ **CalculateModes** (C)
$L \leftarrow$ **Decoding** (I', λ)
return L

Figure 5. Watermark extraction algorithm

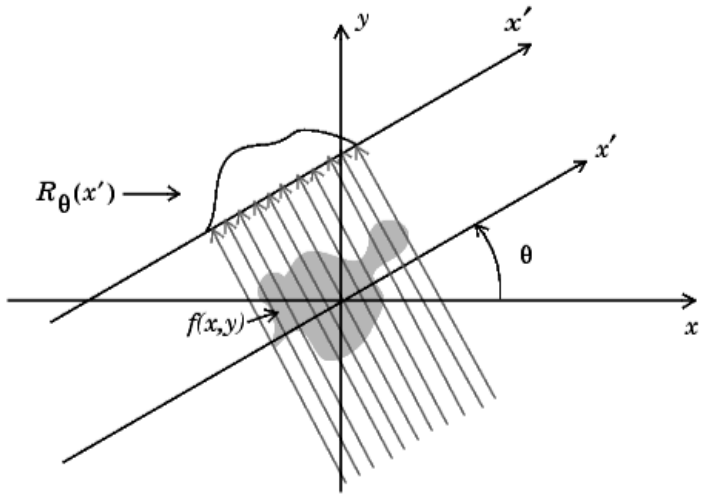

Figure 6. Radon transform

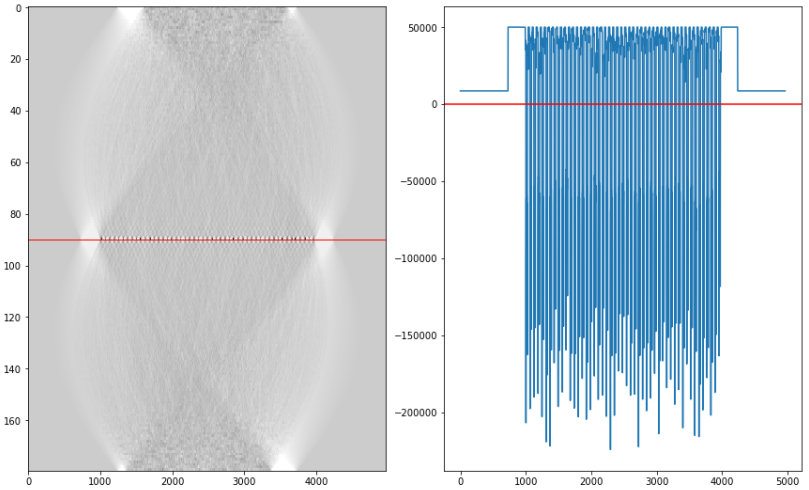

Figure 7. Interline spaces extraction

Watermark embedding capacity I

$$N = \left\lfloor \frac{H - (m_h + m_d)}{\gamma + \beta \cdot \gamma} \right\rfloor \qquad (1)$$

- N – text lines number;
- H – text document height;
- m_t, m_d – top and bottom margin;
- β – interline spacing value;
- γ – font size.

Table 1. The text lines number dependence of page completely filled with text on the text document parameters (A4 page, $H = 297$ mm)

Font size γ (pt)	Interline spacing value β (multiplier)	Text lines number
10	1	63
10	1,25	50
10	1,5	42
12	1	52
12	1,25	42
12	1,5	35
14	1	45
14	1,25	36
14	1,5	30

Watermark embedding capacity II

$$\eta = \lfloor \log_2(\lambda^{N-1} - 1) \rfloor \qquad (2)$$

- λ – embedding components value;
- η – maximum achievable embedding capacity;

Table 2. **The maximum achievable embedding capacity dependence on the text lines number, the embedding components number and a text document parameters**

Text lines number N	Maximum achievable embedding capacity η (bit)						
	$\lambda = 2$	$\lambda = 3$	$\lambda = 4$	$\lambda = 5$	$\lambda = 6$	$\lambda = 7$	$\lambda = 8$
63	62	98	124	143	161	174	186
52	51	80	102	118	132	143	153
45	44	69	88	102	114	123	132
42	41	64	82	95	106	115	123
36	35	55	70	81	91	98	105
30	29	45	58	67	75	81	87

Watermark embedding capacity III

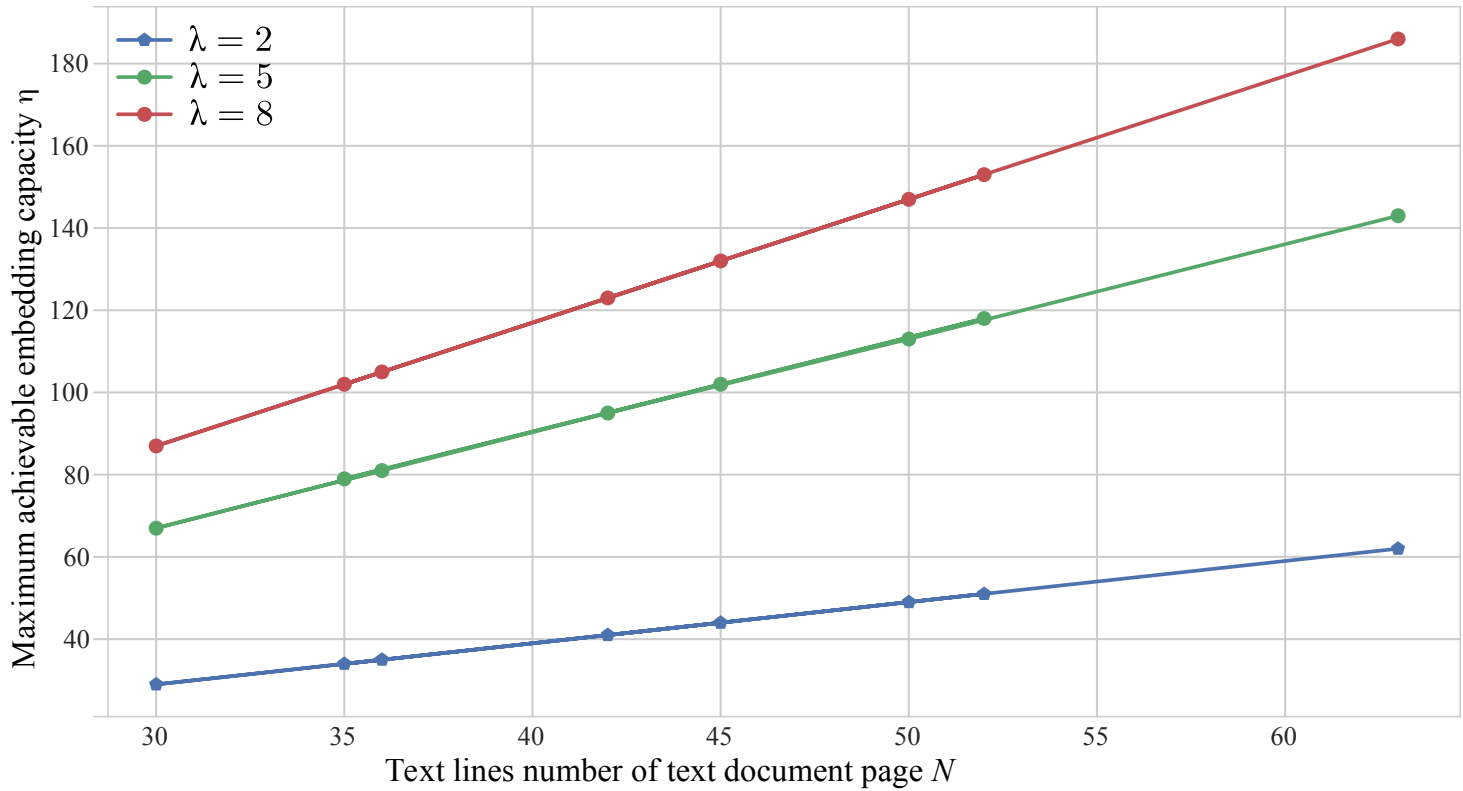

Figure 8. Dependence of the maximum achievable embedding capacity on the number of components used

Analytical expressions for the maximum achievable embedding capacity:

$$f(N)_{\lambda=2} = N - 1$$

$$f(N)_{\lambda=3} = 1{,}59 \cdot N - 2$$

$$f(N)_{\lambda=4} = 2 \cdot N - 2$$

$$f(N)_{\lambda=5} = 2{,}31 \cdot N - 2$$

$$f(N)_{\lambda=6} = 2{,}61 \cdot N - 3$$

$$f(N)_{\lambda=7} = 2{,}81 \cdot N - 3$$

$$f(N)_{\lambda=8} = 3 \cdot N - 3$$

Influence of watermark parameters to embedding capacity

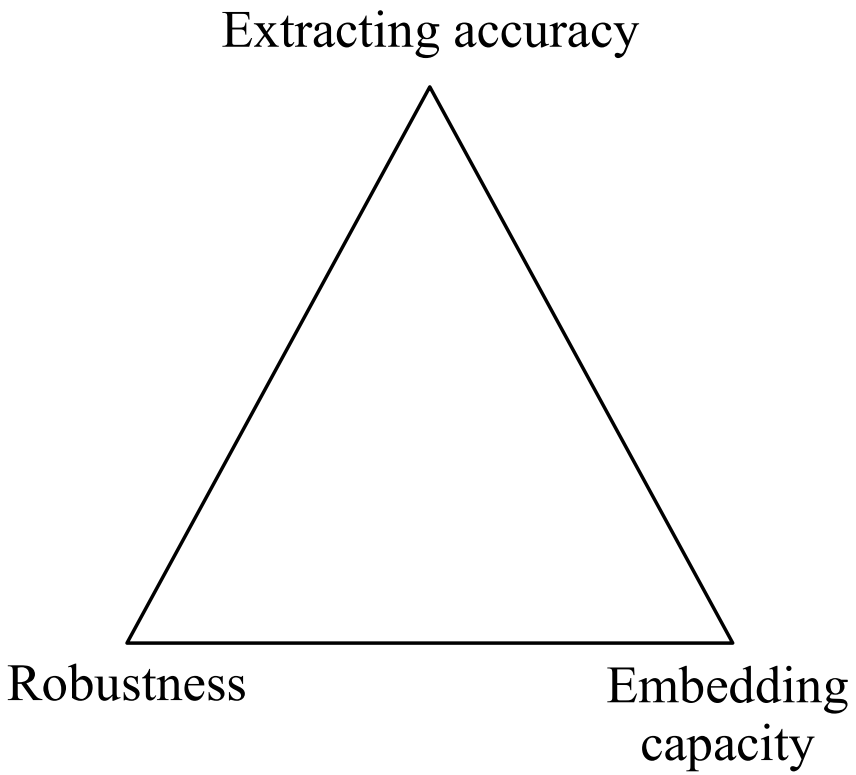

Figure 9. Robust watermark parameters inconsistency

Extracting accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$F - measure = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \quad (4)$$

Robustness:

$$Rob \,|\, \forall Acc(extract(Im)) \geqslant 0{,}90 \quad (5)$$

- $extract$ – extraction function;
- Acc – extracting accuracy.

Embedding capacity:

$$Cap = \eta(TD_O, \lambda) \quad (6)$$

$$embed(L, N, \lambda) \rightarrow \langle Cap, Acc \rangle \quad (7)$$

- L – embedding information;
- TD_O – text document.

Extracting accuracy I (depending on Δ)

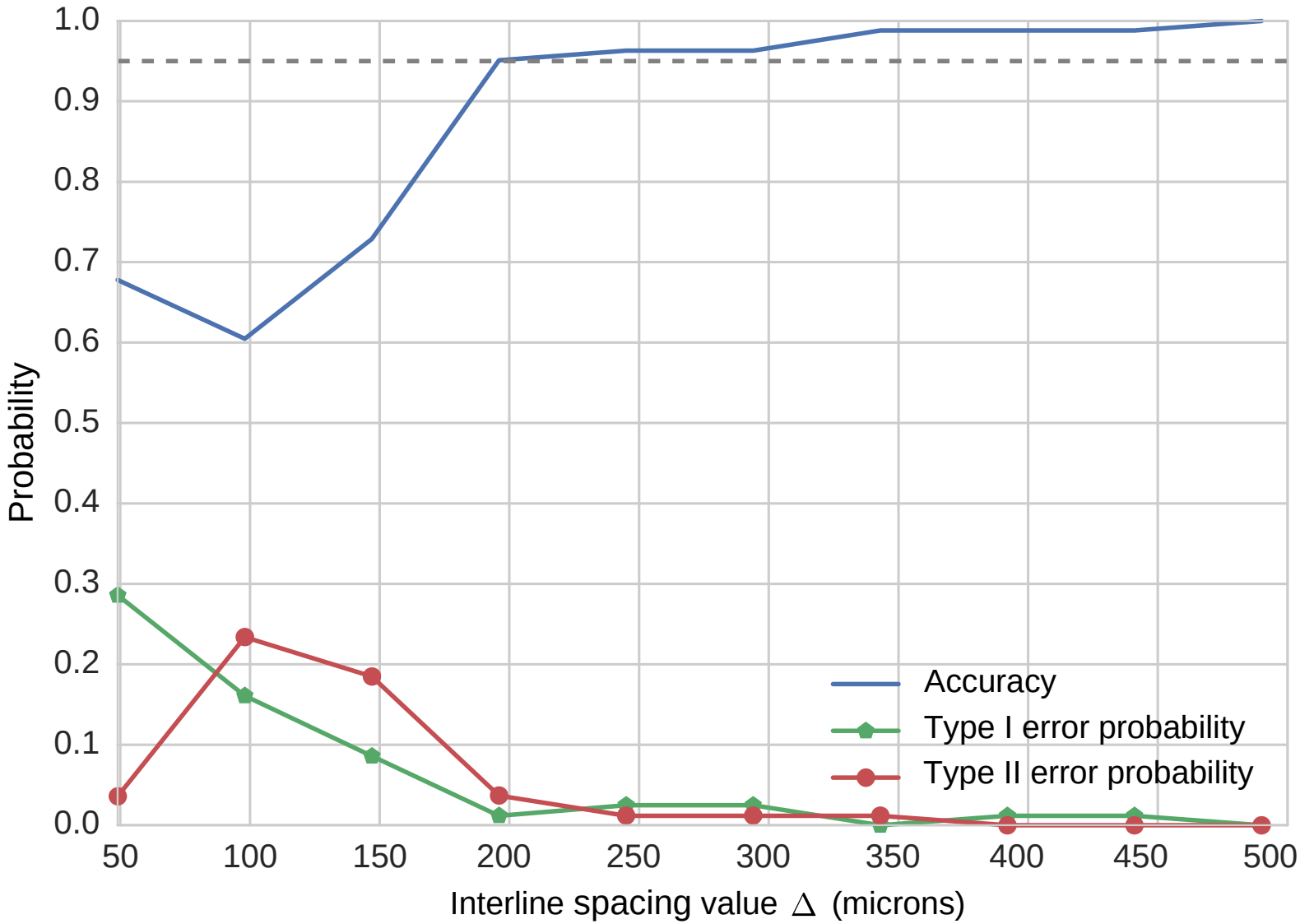

Figure 10. Dependence of the maximum achievable embedding capacity on the number of components used

Extracting accuracy II (depending on DPI)

Table 3. Information extraction from a scanned image with font settings: $\gamma = 14$ пт, $\beta = 1$, $\lambda = 2$, $\Delta = 0{,}05$

Image resolution (DPI)	True positive rate TPR	True negative rate TNR	$Accuracy$	$F - measure$
150	1	0,86	0,90	0,92
200	0,95	0,93	0,94	0,94
300	0,96	0,93	0,95	0,95
400	0,96	0,94	0,95	0,96
600	0,96	0,94	0,95	0,96

Table 4. Information extraction from a scanned image with font settings: $\gamma = 14$ пт, $\beta = 1{,}5$, $\lambda = 2$, $\Delta = 0{,}05$

Image resolution (DPI)	True positive rate TPR	True negative rate TNR	$Accuracy$	$F - measure$
150	0,90	0,83	0,86	0,86
200	0,98	0,88	0,93	0,93
300	0,98	0,92	0,95	0,95
400	0,98	0,93	0,95	0,95
600	1	0,90	0,95	0,95

Extracting accuracy III (depending on λ)

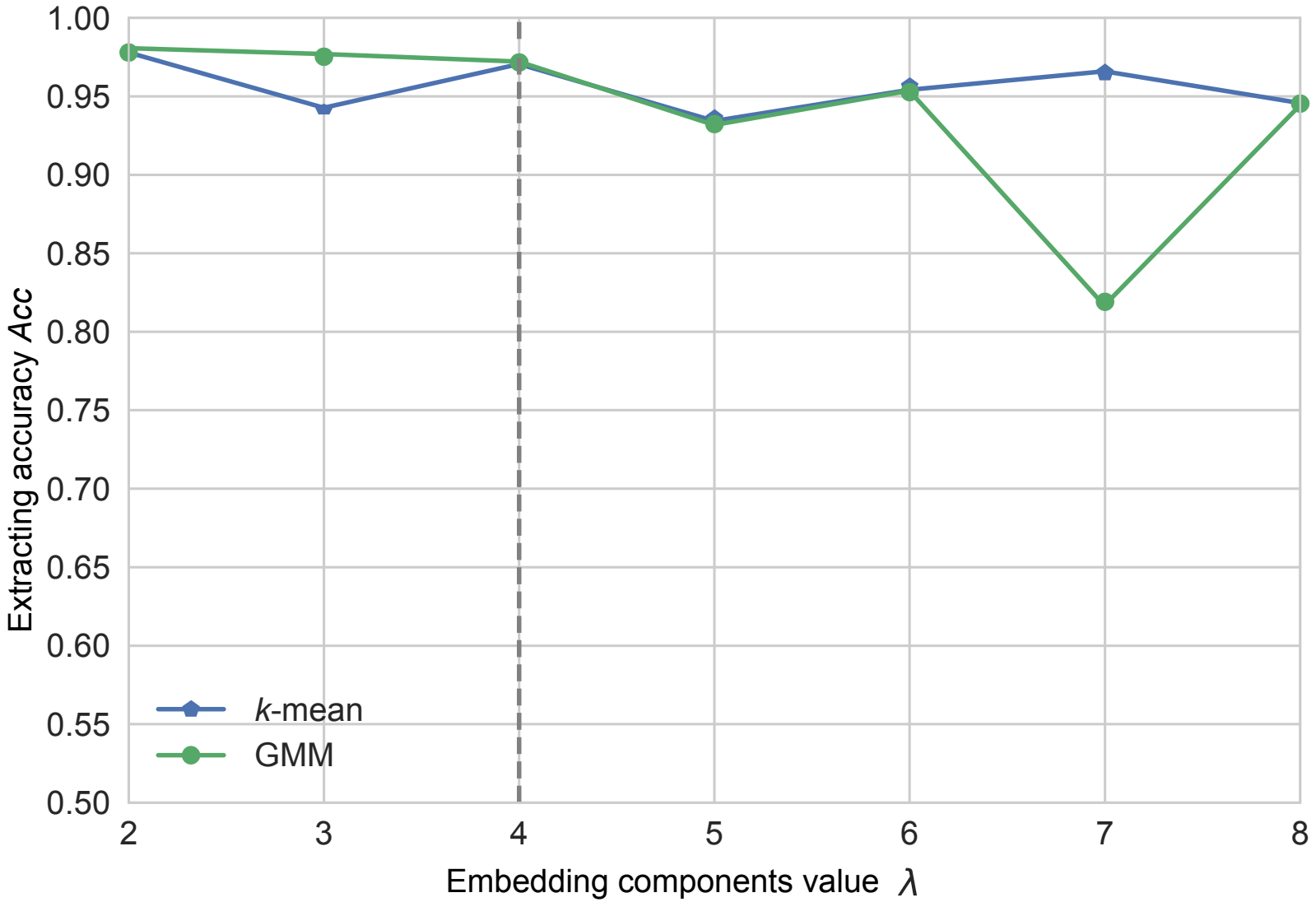

Figure 11. Dependence of the information extraction accuracy on the embedding components number

Extracting accuracy enhancement

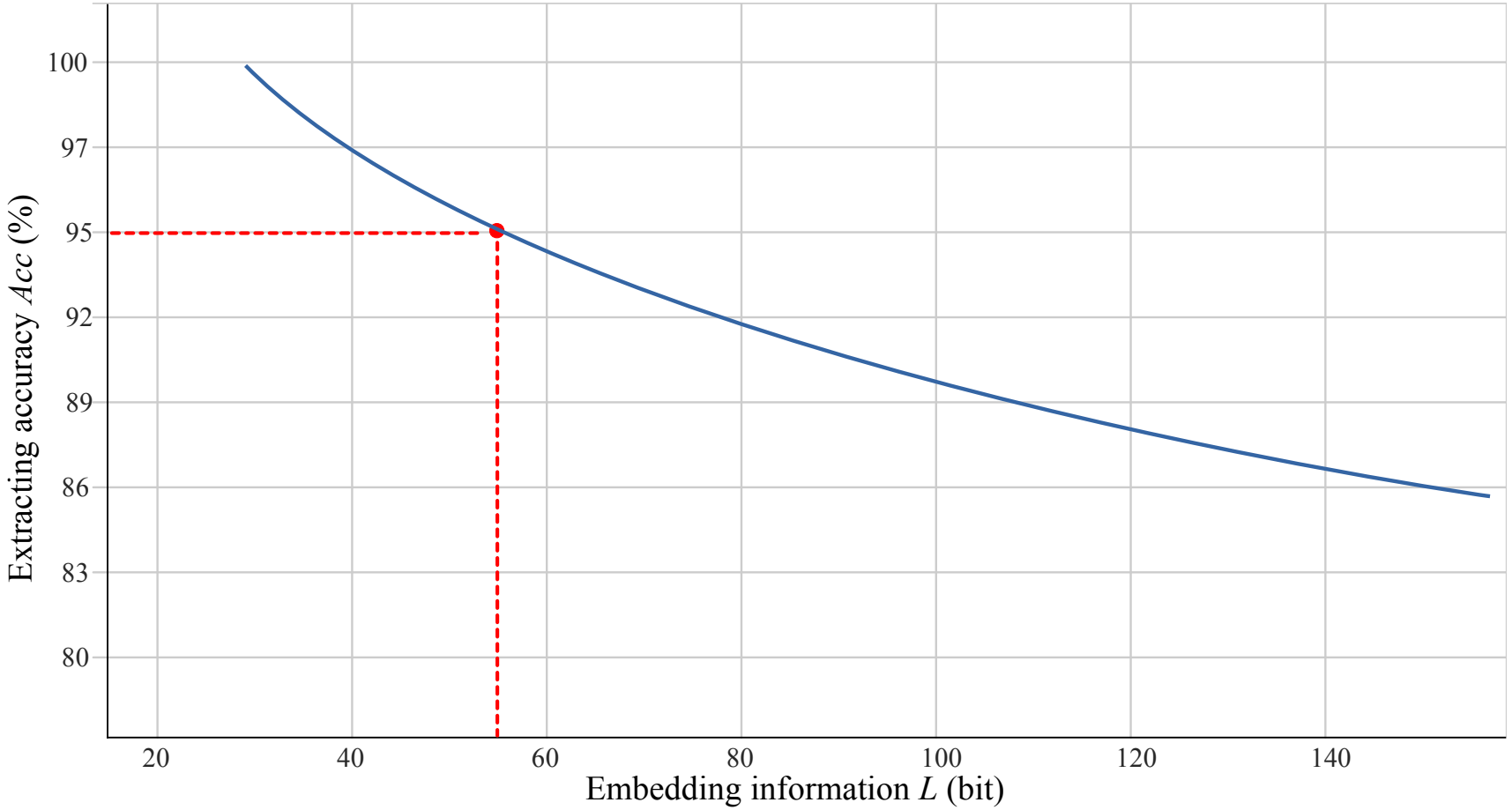

Figure 12. Extracting accuracy dependence on the watermark size (embedding information L)

Hamming code (binary cyclic linear):

- (7,4) – ↑ Acc $+14\%$, ↓ L -40%;
- (15,11) – ↑ Acc $+7\%$, ↓ L -25%.

Reed-Solomon code (nonbinary cyclic linear):

- (7,3) – ↑ Acc $+28\%$, ↓ L -50%;
- (15,11) – ↑ Acc $+18\%$, ↓ L -25%.

Academy of the Federal Guard Service

Estimation of Watermark Embedding Capacity with Line Space Shifting

Alexander Kozachok, Dr. Sci.
Sergey Kopylov

September, 25