NeurIPS 2026 · Poster

FiLM-CAM

Keyed Feature Modulation for Conditional-Access Watermarking

One shared model with key-conditioned watermark decode and remove

Yash Kulthe[1,2] Shruti Agarwal[2] Vishal Asnani[2]
Andrew Gilbert[1] John Collomosse[1,2]

[1] DECaDE, University of Surrey, UK [2] Adobe Research

One watermarked image, two keys



Right key
payload decoded; watermark removable

Wrong key
random bits; watermark survives removal

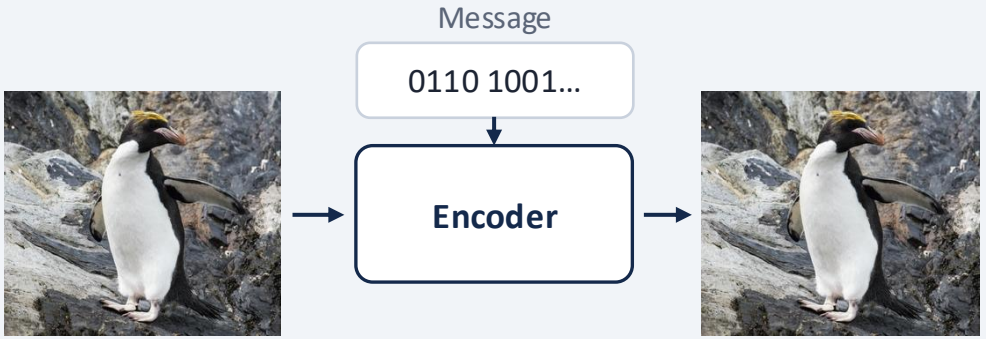
Adobe

CVSSP
Centre for Vision, Speech and Signal Processing

DECaDE

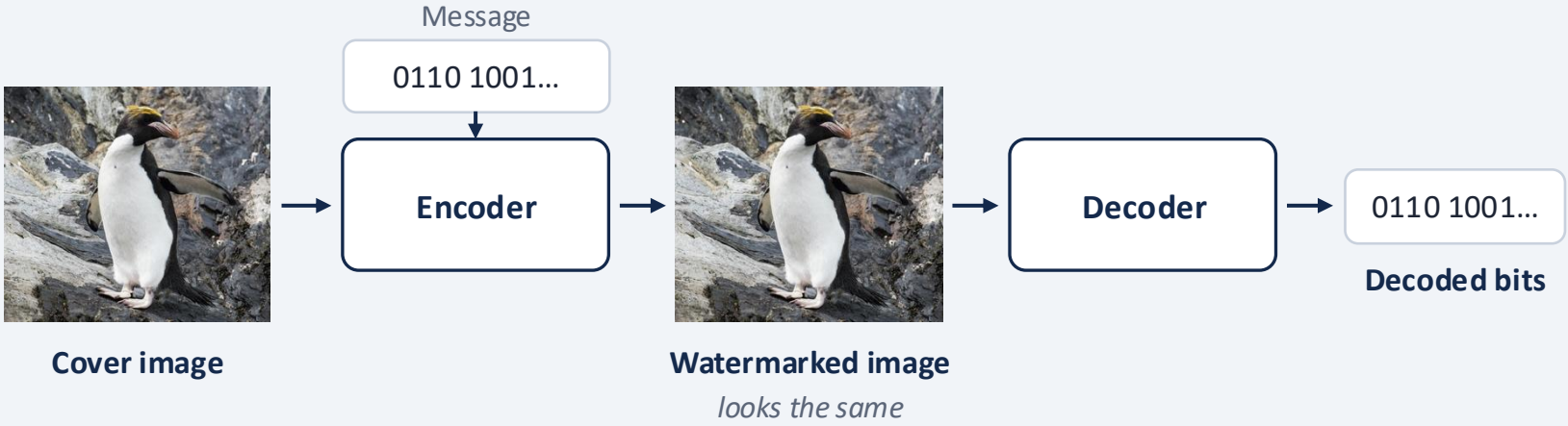
University of Surrey

Invisible watermarking hides a decodable message in an image



Learned encoder–decoder watermarking, e.g. TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025)

Invisible watermarking hides a decodable message in an image



Used to **trace where an image came from** and to **flag AI-generated content**

Learned encoder–decoder watermarking, e.g. TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025)

Open decoders let anyone read, spoof or remove the watermark

Open watermark model



Public decoder → 0110 1001…

Anyone holding the open model can:

- read it
- spoof it
- remove it

The only access control is keeping the model private

Private models today

- Org A → Encoder–decoder A
- Org B → Encoder–decoder B
- Org C → Encoder–decoder C

N organizations → N models to train, ship and maintain

Can one shared model make decoding and removal key-dependent?

Desired behaviour	Correct key
Decode	✓ Payload recovered
Remove	✓ Watermark erased, cover restored

Can one shared model make decoding and removal key-dependent?

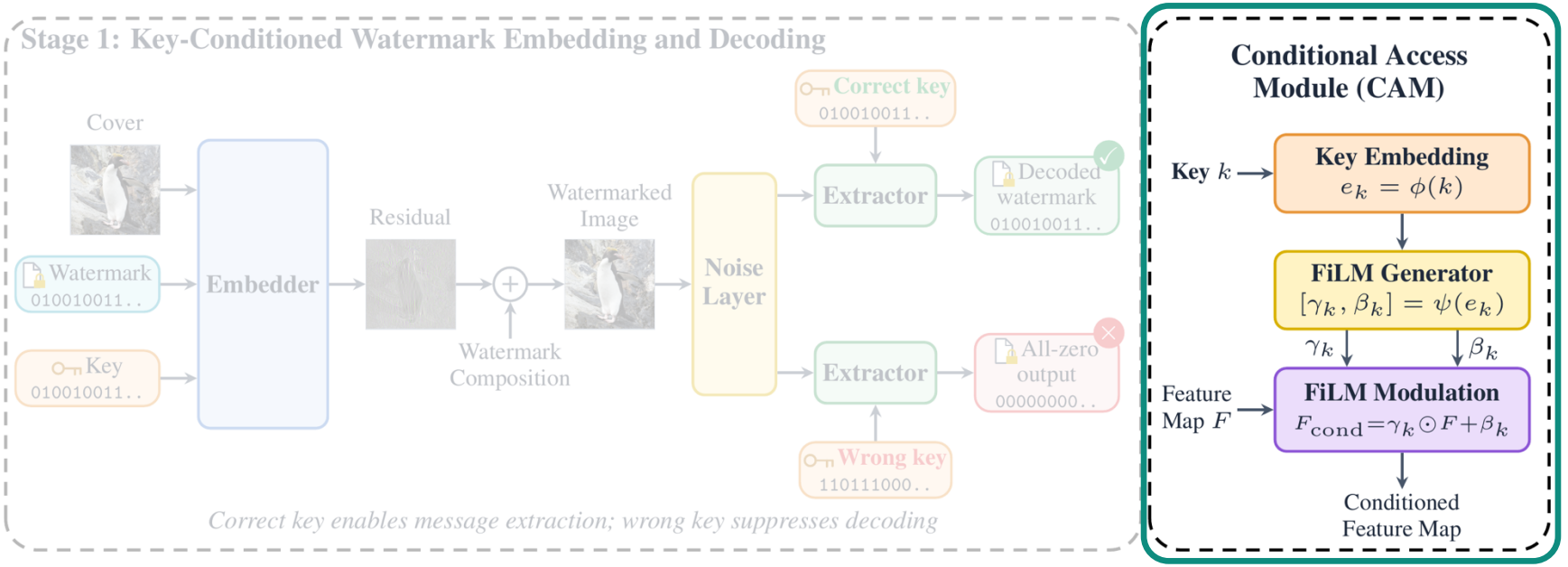
Desired behaviour	Correct key	Wrong key
Decode	Payload recovered	Bits no better than chance (50% bit acc)
Remove	Watermark erased, cover restored	Removal fails; watermark survives

Can one shared model make decoding and removal key-dependent?

Desired behaviour	Correct key	Wrong key
Decode	Payload recovered	Bits no better than chance (50% bit acc)
Remove	Watermark erased, cover restored	Removal fails; watermark survives

Our approach: put the key inside the network, through a Conditional Access Module (CAM) in one shared encoder–decoder.

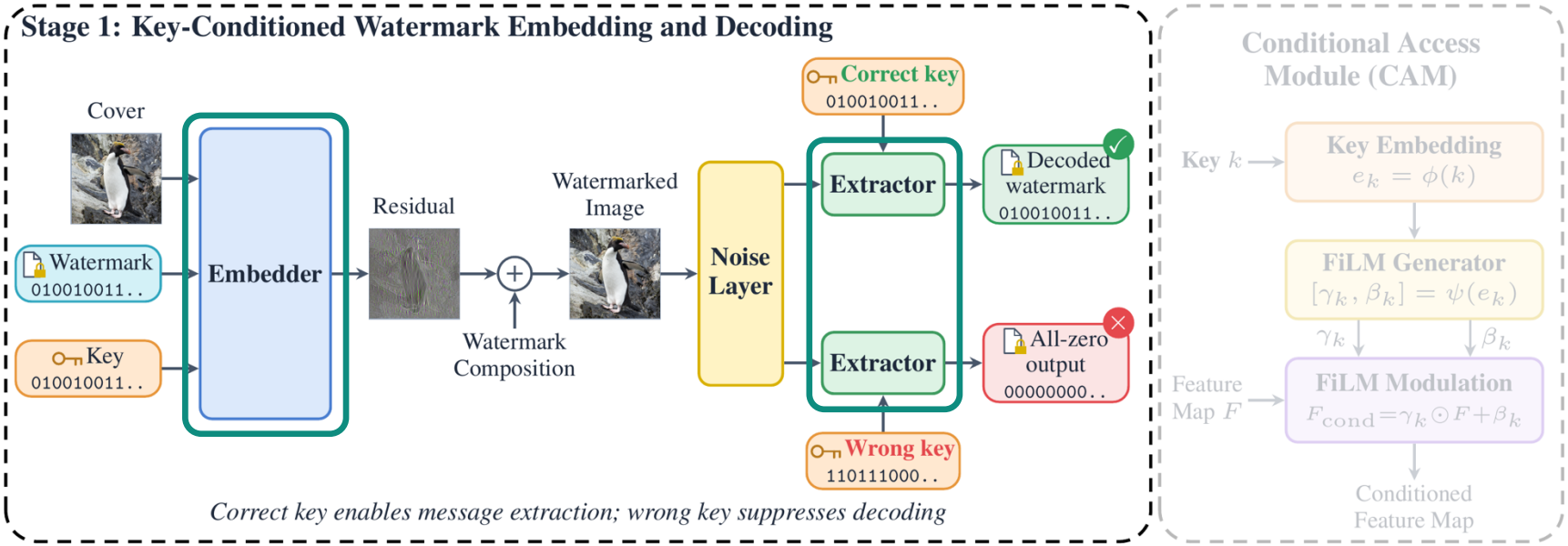
A secret key modulates internal features via FiLM



Training: each correct-key step (recover the watermark) is paired with a wrong-key step (target: all zeros).

Paper Fig. 1 (top), Sec. 3.1–3.2, Sec. 3.6. FiLM: Perez et al. (2018)

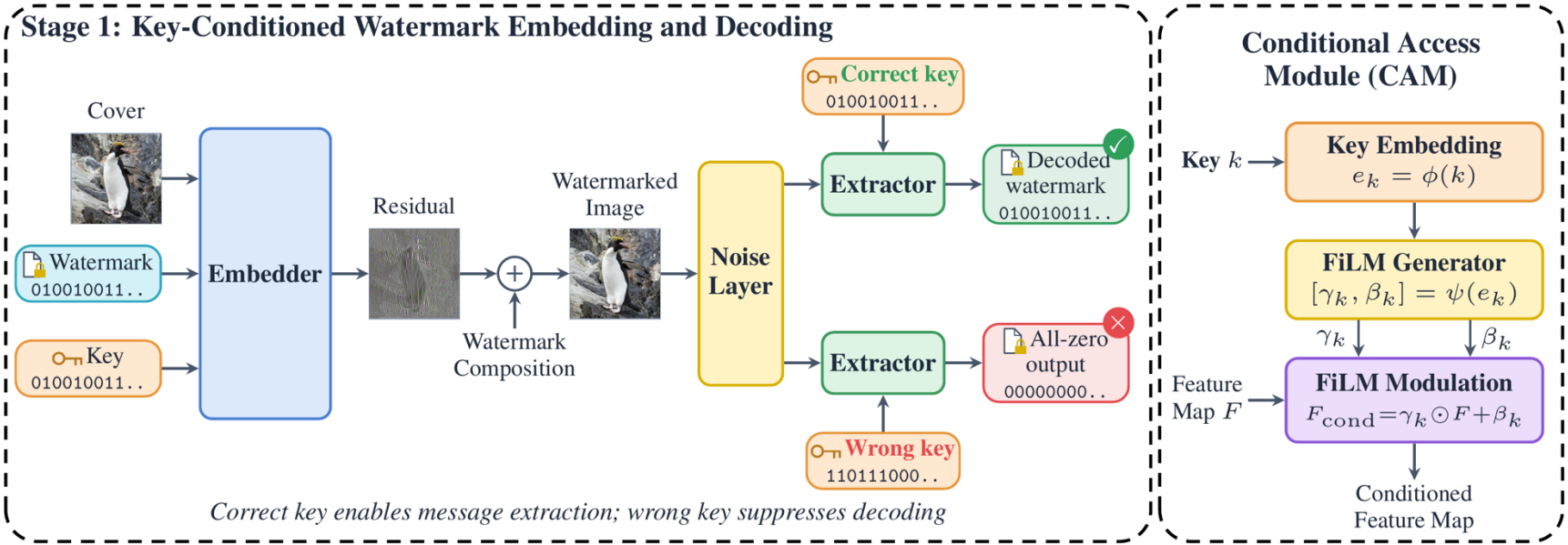
A secret key modulates internal features via FiLM



Training: each correct-key step (recover the watermark) is paired with a wrong-key step (target: all zeros).

Paper Fig. 1 (top), Sec. 3.1–3.2, Sec. 3.6. FiLM: Perez et al. (2018)

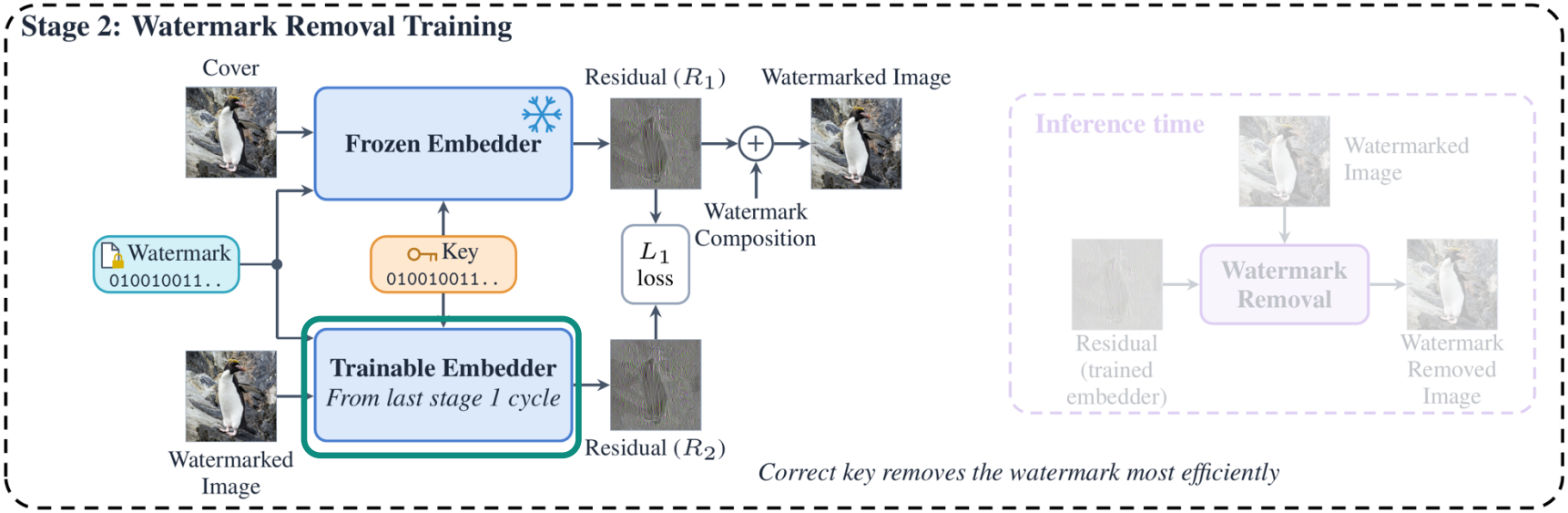
A secret key modulates internal features via FiLM



Training: each correct-key step is paired with a wrong-key step (target: all zeros).

Paper Fig. 1 (top), Sec. 3.1–3.2, Sec. 3.6. FiLM: Perez et al. (2018)

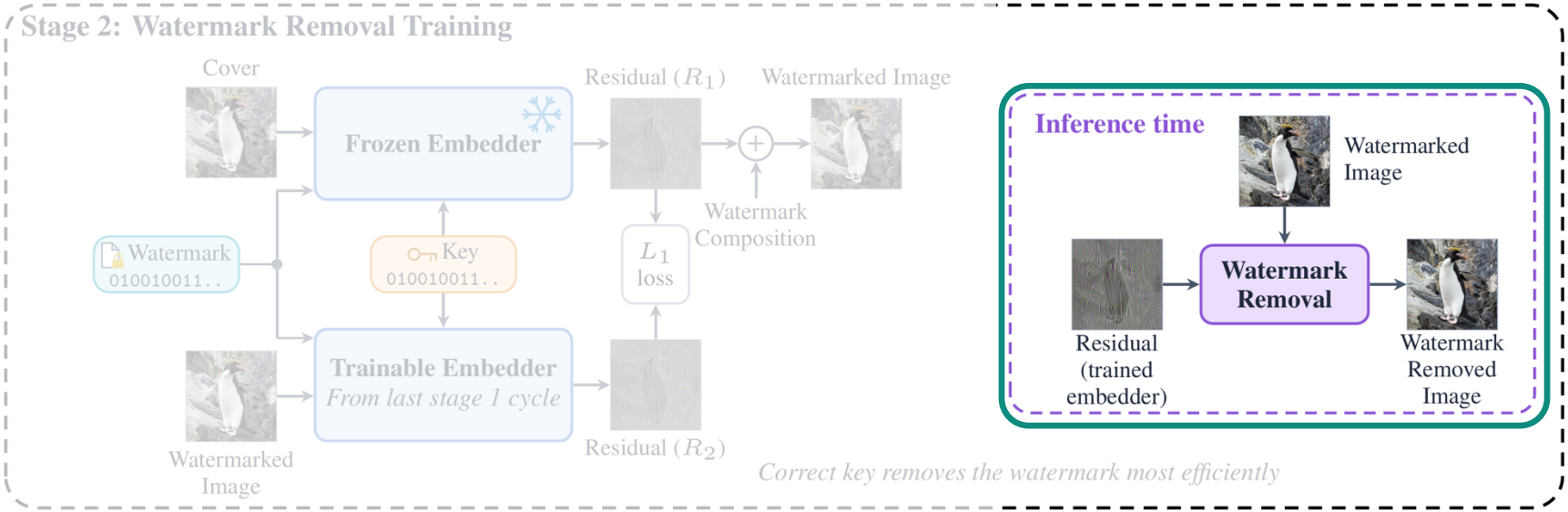
No separate remover: the embedder predicts its own residual



Same network: Stage 2 fine-tunes the embedder to output the residual it embedded.

Paper Fig. 1 (bottom), Sec. 3.3, Sec. 3.5, Algorithm 1; ablation Table 3a

No separate remover: the embedder predicts its own residual

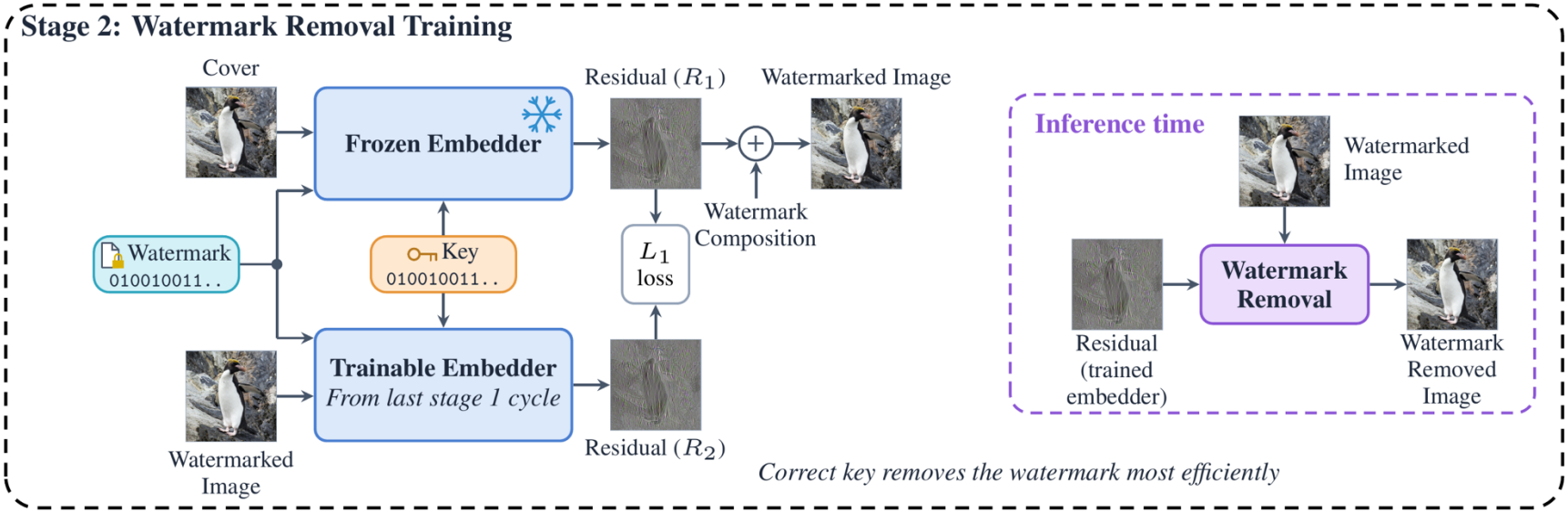


Same network: Stage 2 fine-tunes the embedder to output the residual it embedded.

Logit-space inversion: subtract the residual, refine 20 steps; cleaner than additive (59.3 vs 55.5 dB, COCO).

Paper Fig. 1 (bottom), Sec. 3.3, Sec. 3.5, Algorithm 1; ablation Table 3a

No separate remover: the embedder predicts its own residual



Same network: Stage 2 fine-tunes the embedder to output the residual it embedded.

Logit-space inversion: subtract the residual, refine 20 steps; cleaner than additive (59.3 vs 55.5 dB, COCO).

Paper Fig. 1 (bottom), Sec. 3.3, Sec. 3.5, Algorithm 1; ablation Table 3a

Qualitative results: watermarked image is visually similar to the cover

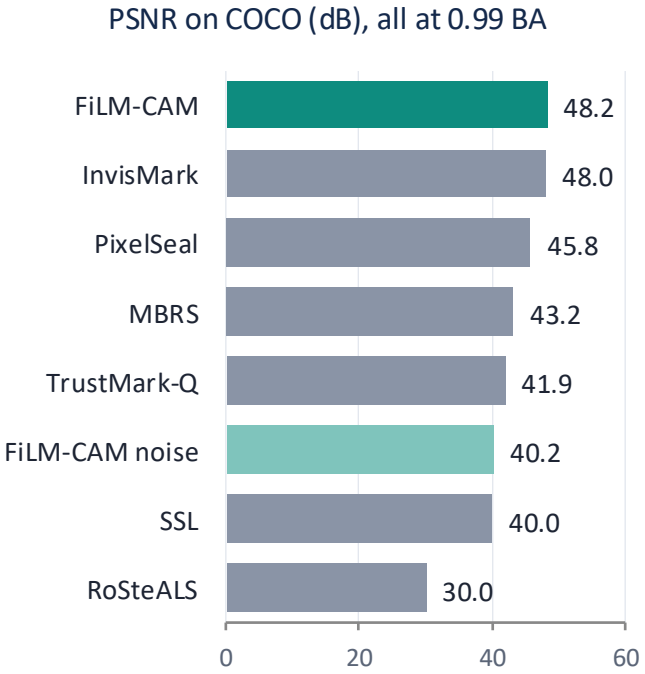


Top: watermarked images. Bottom: amplified $|x_w - x|$, same cover for every method.

- Watermarked image is visually similar to the cover

Paper Fig. 2 (residuals amplified for visibility)

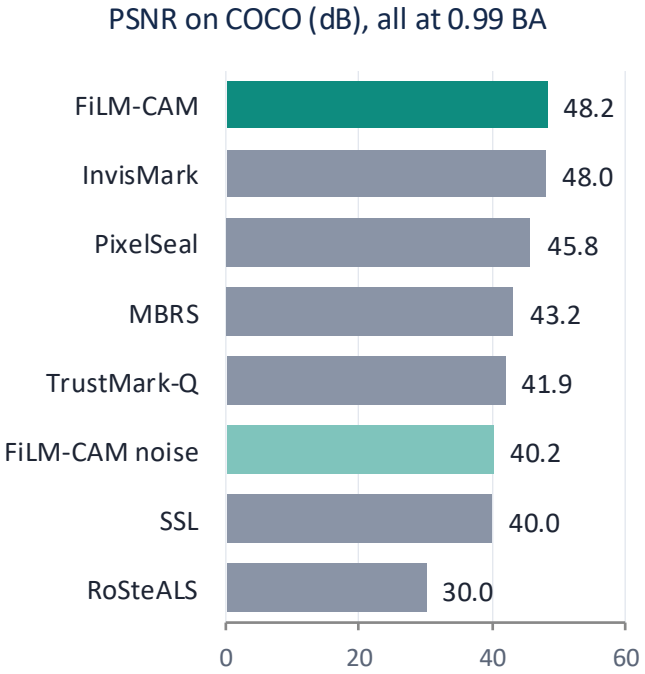
Correct key matches baseline quality; wrong key gives chance



Method	Payload	COCO		DIV2K		SA-1B	
		BA	PSNR	BA	PSNR	BA	PSNR
TrustMark-Q	100	0.99	41.9	0.99	40.9	0.99	41.3
InvisMark	100	0.99	48.0	0.99	46.5	0.99	46.2
PixelSeal	256	0.99	45.8	0.99	46.0	0.99	46.5
RoSteALS	100	0.99	30.0	0.98	28.2	0.99	28.6
SSL	100	0.99	40.0	0.99	40.0	0.99	40.0
MBRS	256	0.99	43.2	0.99	42.2	0.99	42.5
Key-based FNNS	1 bpp	0.99	40.4	0.99	33.9	–	–
FiLM-CAM, right key	100†	0.99	**48.2**	0.99	**46.7**	0.99	**46.6**
FiLM-CAM, wrong key	100†	**0.53**	–	**0.54**	–	**0.53**	–
FiLM-CAM noise, right key	100†	0.99	40.2	0.99	40.1	0.99	40.1
FiLM-CAM noise, wrong key	100†	**0.55**	–	**0.55**	–	**0.55**	–

Paper Table 1 (SSIM, LPIPS in paper). † plus a separate key. Baselines: TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025), PixelSeal (Souček et al., 2025), RoSteALS (Bui et al., 2023), SSL (Fernandez et al., 2022), MBRS (Jia et al., 2021), FNNS (Luo et al., 2023)

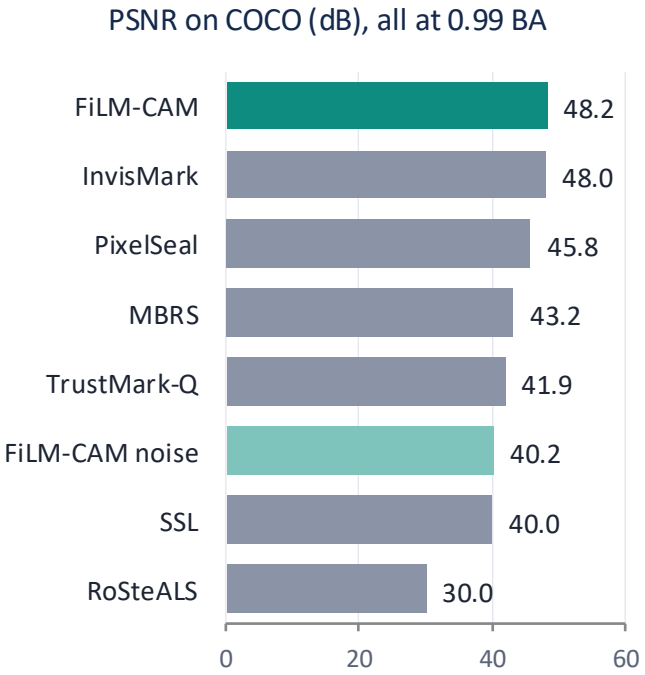
Correct key matches baseline quality; wrong key gives chance



Method	Payload	COCO		DIV2K		SA-1B	
		BA	PSNR	BA	PSNR	BA	PSNR
TrustMark-Q	100	0.99	41.9	0.99	40.9	0.99	41.3
InvisMark	100	0.99	48.0	0.99	46.5	0.99	46.2
PixelSeal	256	0.99	45.8	0.99	46.0	0.99	46.5
RoSteALS	100	0.99	30.0	0.98	28.2	0.99	28.6
SSL	100	0.99	40.0	0.99	40.0	0.99	40.0
MBRS	256	0.99	43.2	0.99	42.2	0.99	42.5
Key-based FNNS	1 bpp	0.99	40.4	0.99	33.9	–	–
FiLM-CAM, right key	100†	0.99	**48.2**	0.99	**46.7**	0.99	**46.6**
FiLM-CAM, wrong key	100†	**0.53**	–	**0.54**	–	**0.53**	–
FiLM-CAM noise, right key	100†	0.99	40.2	0.99	40.1	0.99	40.1
FiLM-CAM noise, wrong key	100†	**0.55**	–	**0.55**	–	**0.55**	–

Paper Table 1 (SSIM, LPIPS in paper). † plus a separate key. Baselines: TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025), PixelSeal (Souček et al., 2025), RoSteALS (Bui et al., 2023), SSL (Fernandez et al., 2022), MBRS (Jia et al., 2021), FNNS (Luo et al., 2023)

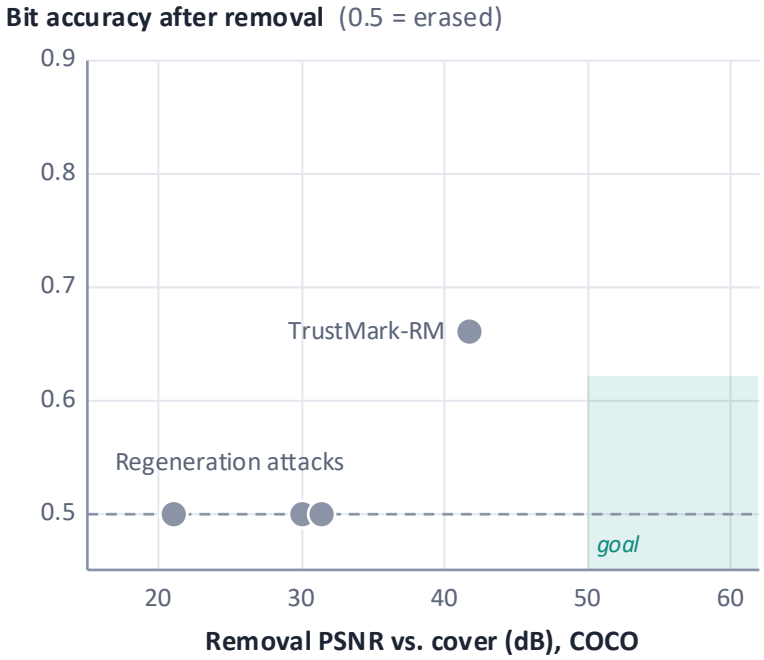
Correct key matches baseline quality; wrong key gives chance



Method	Payload	COCO		DIV2K		SA-1B	
		BA	PSNR	BA	PSNR	BA	PSNR
TrustMark-Q	100	0.99	41.9	0.99	40.9	0.99	41.3
InvisMark	100	0.99	48.0	0.99	46.5	0.99	46.2
PixelSeal	256	0.99	45.8	0.99	46.0	0.99	46.5
RoSteALS	100	0.99	30.0	0.98	28.2	0.99	28.6
SSL	100	0.99	40.0	0.99	40.0	0.99	40.0
MBRS	256	0.99	43.2	0.99	42.2	0.99	42.5
Key-based FNNS	1 bpp	0.99	40.4	0.99	33.9	–	–
FiLM-CAM, right key	100†	0.99	**48.2**	0.99	**46.7**	0.99	**46.6**
FiLM-CAM, wrong key	100†	**0.53**	–	**0.54**	–	**0.53**	–
FiLM-CAM noise, right key	100†	0.99	40.2	0.99	40.1	0.99	40.1
FiLM-CAM noise, wrong key	100†	**0.55**	–	**0.55**	–	**0.55**	–

Paper Table 1 (SSIM, LPIPS in paper). † plus a separate key. Baselines: TrustMark (Bui et al., 2025), InvisMark (Xu et al., 2025), PixelSeal (Souček et al., 2025), RoSteALS (Bui et al., 2023), SSL (Fernandez et al., 2022), MBRS (Jia et al., 2021), FNNS (Luo et al., 2023)

The encoder removes the mark better than baselines

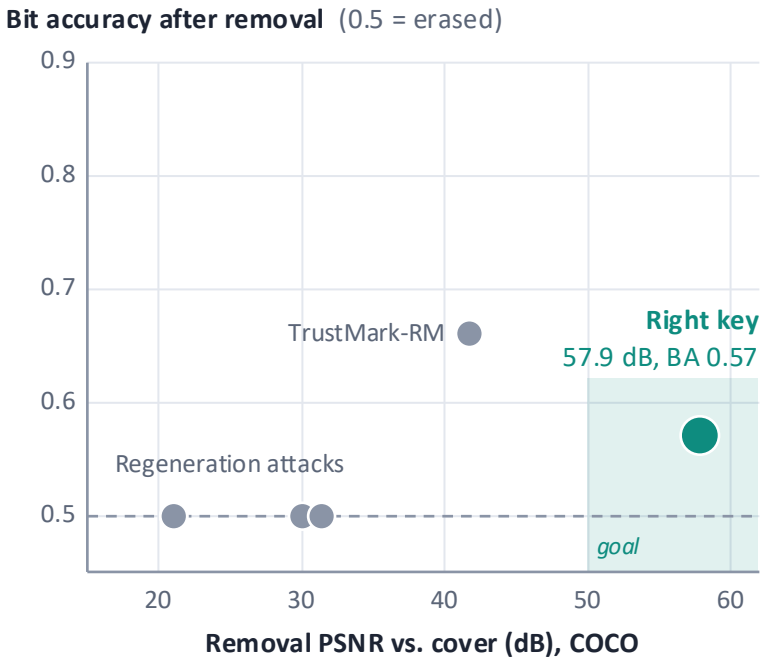


Method	COCO		DIV2K		SA-1B		ms
	PSNR	BA	PSNR	BA	PSNR	BA	
TrustMark-RM	41.7	0.66	40.8	0.74	41.08	0.72	35.9
CtrlRegen+	21.1	0.50	20.0	0.51	20.0	0.50	1976
Diffusion attack	21.0	0.50	22.4	0.51	22.0	0.50	255.6
VAE-Bmshj2018	30.0	0.50	25.3	0.51	25.5	0.50	4.00
VAE-Cheng2020	31.4	0.50	26.4	0.51	26.6	0.50	14.46
FiLM-CAM, right key	**57.9**	0.57	**54.0**	0.58	**54.0**	0.58	5.06
FiLM-CAM, wrong key	48.0	**0.81**	46.6	**0.79**	46.7	**0.78**	5.06

PSNR and BA measured after removal. BA near 0.5 = watermark erased; with a wrong key it should stay high.

Paper Table 2, noise-trained model. Attacks: Bui et al. 2025; Liu et al. 2025; Zhao et al. 2024; Ballé et al. 2018; Cheng et al. 2020

The encoder removes the mark better than baselines

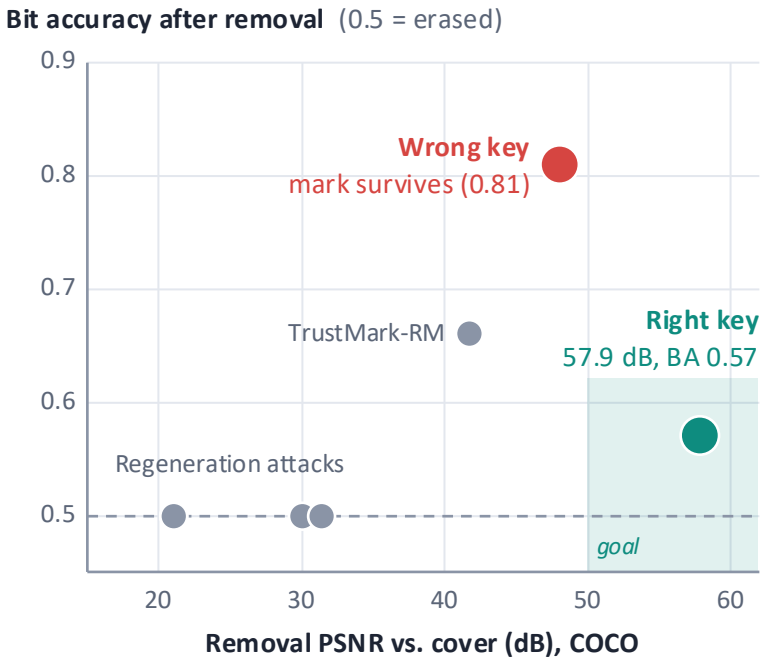


Method	COCO		DIV2K		SA-1B		ms
	PSNR	BA	PSNR	BA	PSNR	BA	
TrustMark-RM	41.7	0.66	40.8	0.74	41.08	0.72	35.9
CtrlRegen+	21.1	0.50	20.0	0.51	20.0	0.50	1976
Diffusion attack	21.0	0.50	22.4	0.51	22.0	0.50	255.6
VAE-Bmshj2018	30.0	0.50	25.3	0.51	25.5	0.50	4.00
VAE-Cheng2020	31.4	0.50	26.4	0.51	26.6	0.50	14.46
FiLM-CAM, right key	**57.9**	0.57	**54.0**	0.58	**54.0**	0.58	5.06
FiLM-CAM, wrong key	48.0	**0.81**	46.6	**0.79**	46.7	**0.78**	5.06

PSNR and BA measured after removal. BA near 0.5 = watermark erased; with a wrong key it should stay high.

Paper Table 2, noise-trained model. Attacks: Bui et al. 2025; Liu et al. 2025; Zhao et al. 2024; Ballé et al. 2018; Cheng et al. 2020

The encoder removes the mark better than baselines

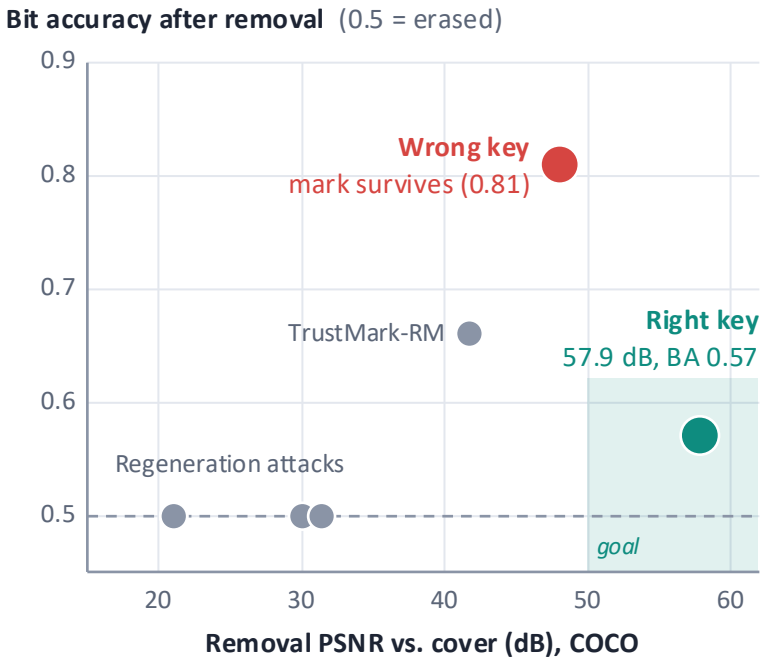


Method	COCO		DIV2K		SA-1B		ms
	PSNR	BA	PSNR	BA	PSNR	BA	
TrustMark-RM	41.7	0.66	40.8	0.74	41.08	0.72	35.9
CtrlRegen+	21.1	0.50	20.0	0.51	20.0	0.50	1976
Diffusion attack	21.0	0.50	22.4	0.51	22.0	0.50	255.6
VAE-Bmshj2018	30.0	0.50	25.3	0.51	25.5	0.50	4.00
VAE-Cheng2020	31.4	0.50	26.4	0.51	26.6	0.50	14.46
FiLM-CAM, right key	**57.9**	0.57	**54.0**	0.58	**54.0**	0.58	5.06
FiLM-CAM, wrong key	48.0	**0.81**	46.6	**0.79**	46.7	**0.78**	5.06

PSNR and BA measured after removal. BA near 0.5 = watermark erased; with a wrong key it should stay high.

Paper Table 2, noise-trained model. Attacks: Bui et al. 2025; Liu et al. 2025; Zhao et al. 2024; Ballé et al. 2018; Cheng et al. 2020

The encoder removes the mark better than baselines



Method	COCO		DIV2K		SA-1B		ms
	PSNR	BA	PSNR	BA	PSNR	BA	
TrustMark-RM	41.7	0.66	40.8	0.74	41.08	0.72	35.9
CtrlRegen+	21.1	0.50	20.0	0.51	20.0	0.50	1976
Diffusion attack	21.0	0.50	22.4	0.51	22.0	0.50	255.6
VAE-Bmshj2018	30.0	0.50	25.3	0.51	25.5	0.50	4.00
VAE-Cheng2020	31.4	0.50	26.4	0.51	26.6	0.50	14.46
FiLM-CAM, right key	**57.9**	0.57	**54.0**	0.58	**54.0**	0.58	5.06
FiLM-CAM, wrong key	48.0	**0.81**	46.6	**0.79**	46.7	**0.78**	5.06

PSNR and BA measured after removal. BA near 0.5 = watermark erased; with a wrong key it should stay high.

Paper Table 2, noise-trained model. Attacks: Bui et al. 2025; Liu et al. 2025; Zhao et al. 2024; Ballé et al. 2018; Cheng et al. 2020

Conclusions

1 Key-conditioned watermarking: a secret key conditions one network via FiLM, so organizations don't need separate private models.

2 Conditionality: decoding and removal both need the key. A wrong key gives chance (≈ 0.5) and cannot remove the watermark; the right key decodes at 0.99 bit acc.

3 Removal: no separate remover needed. The encoder itself removes the watermark, with higher quality than removal baselines (57.9 vs ≤ 41.7 dB, noise-trained).

Limitation: an attacker without the key can still destroy the watermark (for example, by watermarking over it), but cannot restore a high-quality image.

Project page
yash-research.github.io/filmcam

y.kulthe@surrey.ac.uk

Come and see us at the poster

References

[1] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In International Conference on Learning Representations, 2018.

[2] Tu Bui, Shruti Agarwal, Ning Yu, and John Collomosse. Rosteals: Robust steganography using autoencoder latent space. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 933–942. IEEE, 2023.

[3] Tu Bui, Shruti Agarwal, and John Collomosse. Trustmark: Robust watermarking and watermark removal for arbitrary resolution images. In 2025 IEEE/CVF International Conference on Computer Vision (ICCV), pages 18629–18639. IEEE, 2025.

[4] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 7939–7948, 2020.

[5] Pierre Fernandez, Alexandre Sablayrolles, Teddy Furon, Hervé Jégou, and Matthijs Douze. Watermarking images in self-supervised latent spaces. In Proc. ICASSP, pages 3054–3058. IEEE, 2022.

[6] Zhaoyang Jia, Han Fang, and Weiming Zhang. Mbrs: Enhancing robustness of dnn-based watermarking by mini-batch of real and simulated jpeg compression. In Proceedings of the 29th ACM International Conference on Multimedia, pages 41–49, 2021.

References (continued)

[7] Yepeng Liu, Yiren Song, Hai Ci, Yu Zhang, Haofan Wang, Mike Zheng Shou, and Yuheng Bu. Image watermarks are removable using controllable regeneration from clean noise. In International Conference on Learning Representations, volume 2025, pages 87310–87327, 2025.

[8] Zicong Luo, Sheng Li, Guobiao Li, Zhenxing Qian, and Xinpeng Zhang. Securing fixed neural network steganography. In Proceedings of the 31st ACM International Conference on Multimedia, pages 7943–7951, 2023.

[9] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018.

[10] Tomáš Souček, Pierre Fernandez, Hady Elsahar, Sylvestre-Alvise Rebuffi, Valeriu Lacatusu, Tuan Tran, Tom Sander, and Alexandre Mourachko. Pixelseal: Adversarial-only training for invisible image and video watermarking. arXiv preprint arXiv:2512.16874, 2025.

[11] Rui Xu, Mengya Hu, Deren Lei, Yaxi Li, David Lowe, Alex Gorevski, Mingyu Wang, Emily Ching, and Alex Deng. Invismark: Invisible and robust watermarking for ai-generated image provenance. In 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pages 909–918. IEEE, 2025.

[12] Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai. In Advances in Neural Information Processing Systems (NeurIPS), 2024.