A. Full Experimental Setup

(1) Image Generation Settings. Table 3 shows detailed generation settings for the models (*LlamaGen* [29][4] and *RAR-XL* [44][5]) used in this paper. They align with the optimal settings reported in the corresponding original works for each model.

	LlamaGen (GPT-B)	**LlamaGen (GPT-L)**	**RAR-XL**
Resolution	256×256	384×384	256×256
$\|\mathbb{V}\|$	16384	16384	1024
CFG scale	2.0	2.0	6.9
CFG pow	–	–	1.5
top-k	$\|\mathbb{V}\|$	$\|\mathbb{V}\|$	$\|\mathbb{V}\|$
top-p	1.0	1.0	1.0
temperature	1.0	1.0	1.02

Table 3. Model and generation details. $|\mathbb{V}|$ is the vocabulary size of the VQ-VAE in use.

(2) Baseline Settings. The baseline methods *DWT-DCT-SVD* [25] and *RivaGAN* [45] are taken from a commonly used implementation of post-hoc watermarking schemes[6]. For *TrustMark* [2] and *SSL Watermarking* [10] their respective repositories[7][8] are used. For DWT-DCT-SVD, 64 random bits are encoded for each image. For RivaGAN, 32 random bits are encoded for each image as this is the maximum supported by the implementation. Detection is done by extracting bits from images and assessing the ratio of matching bits. For the concurrently developed AR baselines *IndexMark* [33] and WMAR [16], we use their respective implementations[9][10]. For all methods, with the exception of TrustMark, we determine AUC and TPR empirically by calculating ROC against unwatermarked images. For TrustMark, we use their proposed default configuration Q which yields an optimal trade-off between quality and watermark robustness. Robustness is evaluated directly by using the built-in detection routine, since decoded bits are only exposed once detection is successful.

(4) Perturbation Sets and Visual Examples. We use two perturbation sets during evaluation. *Perturbation Set A* and *Perturbation Set B* are defined in Table 4. Set A includes weaker, more realistic perturbations and has been used in early experiments and for prefix tuning. Set B includes strong perturbations and is used in the main paper to push the watermarking methods to their limits. Note that for the color jitter perturbations, values are sampled randomly in the range specified by maximum intensity given in the tables. Table 5 shows the perturbation settings used during training of the token and cluster classifier. Figure 6 shows visual examples of the perturbations applied during all experiments. *Brightness*, *contrast*, *hue*, and *saturation* are summarized as *color jitter*. Results for *SD-1.5-AE*, *FLUX.1-AE* and *SD2.1 Regeneration* are averaged and summarized as *regeneration*. The regeneration attack [46] is performed with default settings[11], i.e. 60 steps of denoising using Stable Diffusion 2.1 [27].

(4) Computing Infrastructrue and Software Details. The experiments were conducted on two servers.
1. AMD EPYC 7452 (32 cores), 4× Nvidia A40 (48GB), 500 GB RAM, running Ubuntu 20.04.6 LTS
2. AMD EPYC 7282 (16 cores), 4× Nvidia A40 (48GB), 500 GB RAM, running Ubuntu 20.04.6 LTS

The exact Python version and package requirements can be found in our project repository.

(5) Number of Samples Used in Experiments. Images are generated for different purposes: (1) unwatermarked images as negative (unwatermarked) class for evaluation, (2) unwatermarked images used for training the token and cluster classifiers,

[4] https://github.com/FoundationVision/LlamaGen
[5] https://github.com/bytedance/1d-tokenizer
[6] https://github.com/ShieldMnt/invisible-watermark
[7] https://github.com/adobe/trustmark
[8] https://github.com/facebookresearch/ssl_watermarking
[9] https://github.com/maifoundations/IndexMark
[10] https://github.com/facebookresearch/wmar
[11] https://github.com/XuandongZhao/WatermarkAttacker

Augmentation Type	Parameter	Set A	Set B
Gaussian noise	σ	0.05	0.2
JPEG compression	Quality	60	20
Gaussian blur	r	2	3
RGB salt & pepper noise	p	0.03	0.1
Random Drop	Ratio	0.3	0.5
Brightness jitter	Max. Intensity	3	4
Contrast jitter	Max. Intensity	1.5	4
Hue jitter	Max. Intensity	0.1	0.5
Saturation jitter	Max. Intensity	2	5
SD1.5-AE	-	-	-
FLUX.1-AE	-	-	-
SD2.1-Regeneration	Denoising Steps	60	60

Table 4. Perturbation sets A and B. Rows show augmentation types, parameter names and their respective values for both sets.

Augmentation Type	Parameter(s)
Gaussian noise	$\sigma = 0.1$
JPEG compression	Quality range: $80 - 20$
Gaussian blur	Max radius $= 2$
RGB salt & pepper noise	Max $p = 0.07$
Brightness jitter	Max. Intensity $= 4$
Contrast jitter	Max. Intensity $= 2$
Hue jitter	Max. Intensity $= 0.1$
Saturation jitter	Max. Intensity $= 2$

Table 5. Data augmentation settings and parameters used during training of the token and cluster classifier.

(3) watermarked images used in robustness study, (4) watermarked images used to measure FID and CLIP score. Below is a breakdown of the number of generated images:

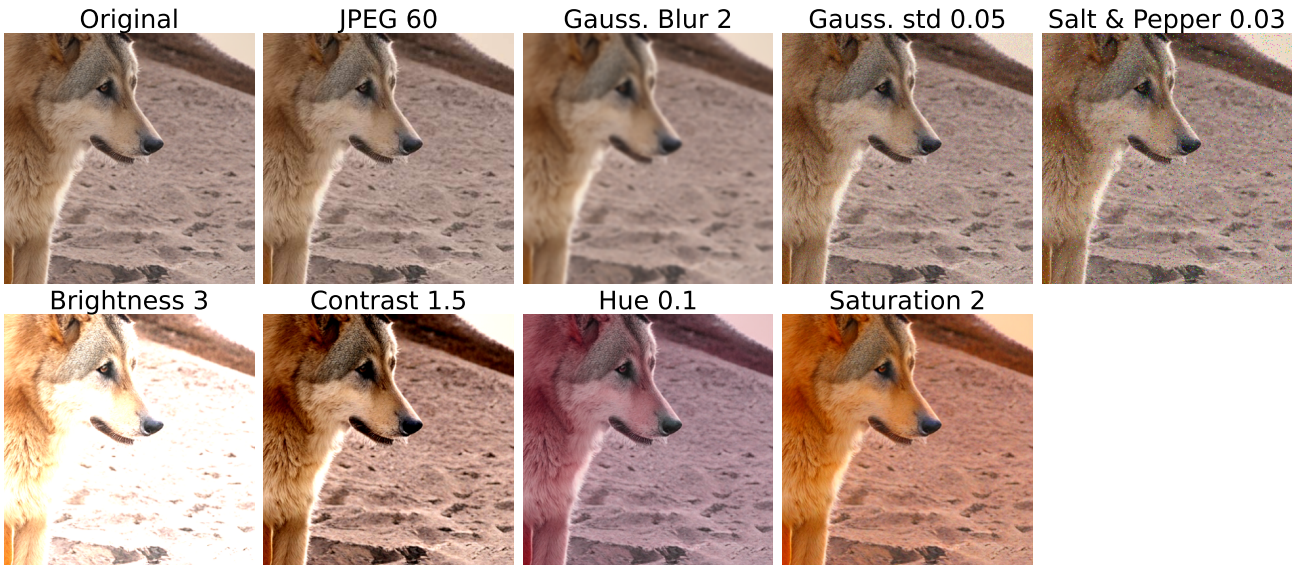
(I) Unwatermarked generated images. For each of the three models (LlamaGen (GPT-B and GPT-L), RAR-XL), we generated 100k images. These images were used for training the cluster and token predictors. In addition, 2,000 unwatermarked generated images were used as negative class for evaluation. The baselines methods (DWT-DCT-SVD, RivaGAN, TrustMark, and SSL watermarking) were applied on top of unwatermarked images to obtain positive samples.

(II) Images used for the tuning of watermarking parameters. For identifying ideal watermarking parameters, we then generated 2,000 samples for each setting combination (number of clusters $k \in [-, 128, 64, 32, 16, 8]$, penalty $\delta \in [2, 5]$, and green token fraction $\gamma \in [0.5, 0.25]$) for each model. This yields 24 settings per model, totaling $24 \times 2,000 \times 3 = 144,000$ images. This is done for 8 different prefixes $\forall\, \kappa \in \{1, \ldots, 8\}$, totaling $144,000 \times 8 = 1,152,000$ images. With this data, prefix tuning was then performed.

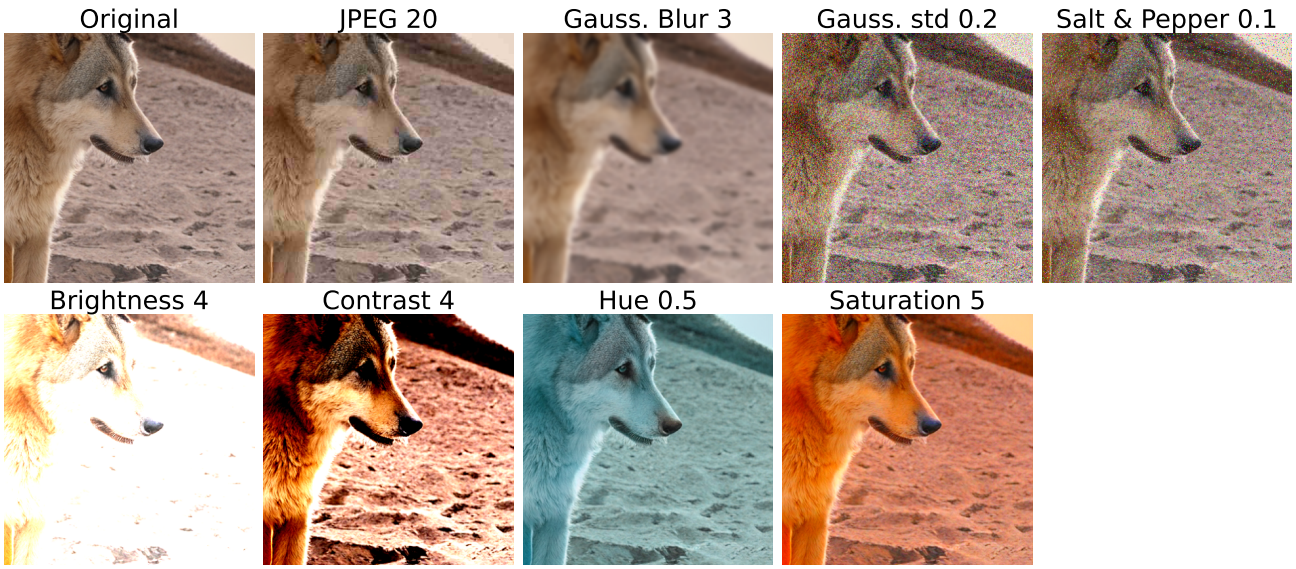
(III) Images used to evaluate FID and CLIP scores. To calculate FID scores and CLIP scores against ImageNet validation set (which contains 50,000 images), 50,000 watermarked images were generated using the best performing hash prefix κ obtained in the previous step for each of the 24 settings and for each of the 3 models, yielding $50,000 \times 24 \times 3 = 3,600,000$ images. Since we kept generation seeds consistent over all experiments, the first 2,000 images for each setting are duplicate to the images used in step II.

(IV) Images for the main robustness study in Section 4. Since we already obtained 50,0000 watermarked images in the previous step for the best performing hash prefix for each setting, we then sliced 2,000 images which are distinct from the images used in step II for all further evaluation in Section 4.

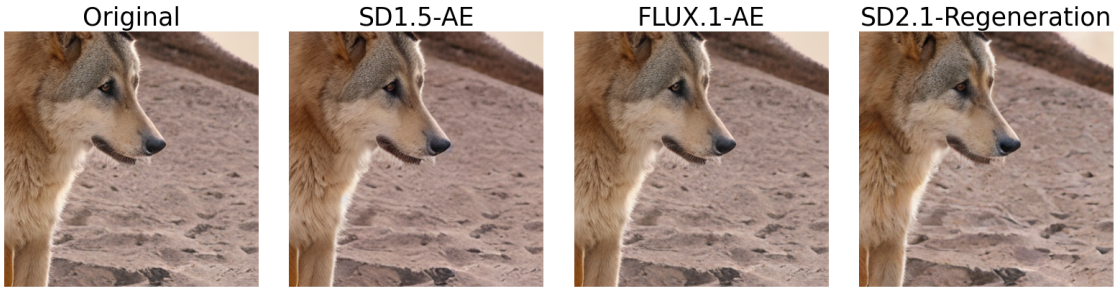
Total. This adds up to $\sim$5M images generated in total.



(a) Examples of **Perturbation Set A** without regeneration attacks (default perturbations used for the ablation and prefix tuning.)



(b) Examples of **Perturbation Set B** without regeneration attacks (strong perturbations used for main results in Table 1)



(c) **Regeneration attacks**. SD1.5-AE and FLUX.1-AE show encode-decode attacks with their respective autoencoders. SD2.1-Regeneration shows the regeneration attack. These attacks are included in both **Perturbation Sets A and B**.

Figure 6. Examples of visual perturbations.

B. Full Experimental Results

(1) Detailed Results Used of Prefix Tuning. The optimal prefixes κ were determined empirically by evaluating 8 possible prefixes $\kappa \in \{1, \ldots, 8\}$ across each setup defined by the combination of the generation model, the addition of the token or cluster classifier, the number of clusters ($k \in 8, 16, 32, 64, 128$), watermark strength ($\delta \in 2, 5$), and fraction of green tokens ($\gamma \in 0.5, 0.25$). The evaluation was performed by first applying the weak perturbation set A on 2,000 images generated with the watermark and calculating ROC characteristics against a set of 2,000 unwatermarked images. The best prefix for each group is determined by averaging the results across all perturbations and choosing the maximum TPR@FPR=1%. These results are shown in Table 7.

To avoid selection bias, the main results shown in Section 4 in the main paper were then performed using the best prefixes, but on a distinct set of 2,000 generated images and using the distinct perturbation set B.

(2) Full Ablation Plots: In Figure 7, the full ablation is provided for the effect of different watermarking settings on the FID. Figure 9 shows full ablation plots for different perturbations on how watermark detection performs under different watermarking settings. Note however, that the perturbation set used (Set A) is different from Set B used in Figure 4.

(3) CLIP scores: We calculate the CLIP scores [14] on 50,000 images across different models and settings by prefixing ImageNet class names corresponding to each generated image class with the prefix "a photo of ". Figure 8 shows the results. The CLIP scores are mainly affected by low numbers of clusters.

(4) Results for Geometric Transformations: Out-of-the-box, token-level watermarks are not expected to be robust against geometric transformations. In the following, we test their robustness to geometric transformations for our 64-cluster method in Tabs. 8 to 10. We then study the effect on the watermark detection by undoing perturbations and by using a synchronization layer. Our analysis consists of three parts.

First, we test against various geometric transformations, with results reported in Tab. 8. As expected, sufficiently strong transformations (x1.5 scaling, 33° rotation, 0.3 perspective, (+6px, +7px) translation, or flipping) essentially destroy the watermark ($<$ 10% TPR, similarly to WMAR without synchronization). **Second**, we apply rotation and scaling, and then apply inverse transformations to see how interpolation artifacts affect watermark strength. Results are reported in Tab. 9. We find that reverted rotation (33°) and scaling (x0.5, x1.5) still lead to 100% TPR. **Finally**, we use SyncSeal [12], a state-of-the-art image synchronization method, and report results in Tab. 10. Just applying SyncSeal on top of our method retains 100% TPR. When SyncSeal is used to restore transformed images to their original state after rotation, scaling, flipping or perspective transformations, our watermark (64 clusters, token predictor) still retains 99+% TPR. Furthermore, the use of clustering significantly improves TPR compared to the case without clustering.

(5) Effect of Cluster Initialization: We also performed a study of the effect of cluster initialization with our 64-cluster variant for RAR-XL. We computed TPR over 6 different cluster initializations for different perturbations in the stronger perturbation set (Set B). We first average over all perturbations in the perturbation set, and subsequently compute the variance over these means. Without the token predictor, the standard deviation is 0.0226, while with the token predictor it is 0.0055. Table 6 reports results for different perturbations in detail. From the results, we observe that for some perturbations (most notably Gaussian blur), the variance is higher, when performance is lower. This may indicate some sensitivity to cluster initialization, although the lack of prefix tuning possibly also has an impact. The variance is lower when combined with the token predictor.

	Clean	G.N 0.2	JPEG20	G.Blur R=3	SP 0.1	Color Jitter
64 clusters	0.999 ± 0.000	0.115 ± 0.048	0.866 ± 0.020	0.296 ± 0.092	0.129 ± 0.041	0.692 ± 0.014
+ token predictor	1.000 ± 0.000	0.902 ± 0.017	0.894 ± 0.024	0.785 ± 0.037	1.000 ± 0.001	0.957 ± 0.005

Table 6. Effect of cluster initialization on robustness of our method, evaluated on different image perturbations. Reported are the non-empirical TPR@FPR=1% values with their standard deviation across multiple cluster initializations.

C. Examples of Generated Images

In Figures 10 to 12, visual examples for both unwatermarked and watermarked images are shown for all models. For watermarked images, we set penalty $\delta = 5$ and green token fraction $\gamma = 0.25$. We show images generated without clustering ($k = 16384$ for LlamaGen GPT-B and GPT-L, $k = 1024$ for RAR-XL) as well as images generated with clustering with $k \in [128, 64, 8]$. While watermarked images retain image quality down to $k = 64$, there is a noticeable decline in image quality for the extreme case of $k = 8$.

D. Security Considerations

In line with previous work on in-generation watermarking [6, 13, 16, 33, 37, 41], we assume a setting in which a service provider exposes an API for watermarked generation and watermark verification. The model (AR backbone, VQ-VAE, and token/cluster predictor) must be kept private to help protect against elaborate removal and forgery attacks [15, 24]. For example, if the VQ-VAE or token/cluster predictor is leaked, the attacker can learn the token patterns associated with the hashing function.

As we established in Section 4.3, our prefix tuning enables the service provider to rule out unfavourable hash prefixes κ empirically. This introduces a potential security concern. As prefix tuning reduces the space of viable prefixes, this could make it easier for an attacker to search for the exact prefix set by the service provider. In our evaluation, we consider a small subset of eight candidates and report the best-performing one. To ensure security in real world deployment, we recommend drawing κ from a sufficiently large space (e.g., 64 bits). Even if only a fraction of prefixes are suitable, the resulting effective key space remains large enough that exhaustive search is infeasible. Therefore, restricting κ to a subset of viable prefixes does not constitute a security risk in practice. Note that such an attack would additionally require access to the encoder.

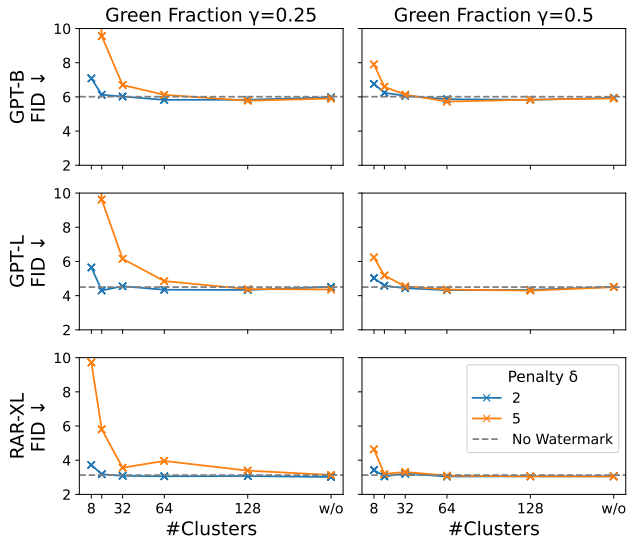


Figure 7. FIDs across models, amount of clusters, penalty δ, and green fraction γ settings.

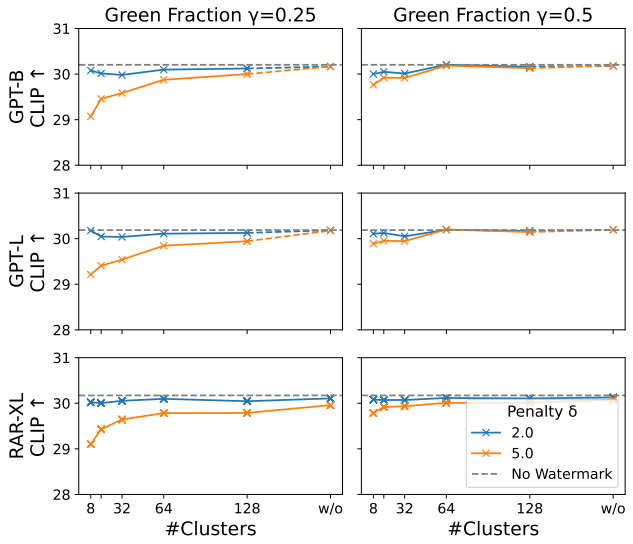


Figure 8. CLIP scores across models, amount of clusters, penalty δ, and green fraction γ settings.

Model	Method	k	δ	γ	κ	FID ↓	Clean	JPEG 60	Gauss. Blur R 2	Gauss. std .05	Salt&Pepper .03	Color Jitter	Regeneration	Average
GPT-B	No WM	-	-	-	-	6.01	– — –	– — –	– — –	– — –	– — –	– — –	– — –	– — –
	Ours (No Clustering)	-	2	0.50	6	5.95	0.998 — 0.989	0.972 — 0.850	0.739 — 0.173	0.898 — 0.582	0.690 — 0.161	0.896 — 0.696	0.870 — 0.640	0.849 — 0.554
				0.25	6	5.97	1.000 — 0.997	0.987 — 0.909	0.771 — 0.257	0.920 — 0.679	0.731 — 0.237	0.914 — 0.738	0.879 — 0.661	0.871 — 0.608
			5	0.50	4	5.91	0.999 — 0.996	0.984 — 0.913	0.780 — 0.256	0.926 — 0.688	0.736 — 0.229	0.906 — 0.731	0.886 — 0.668	0.870 — 0.606
				0.25	4	5.90	0.999 — 0.996	0.995 — 0.967	0.844 — 0.426	0.954 — 0.799	0.783 — 0.356	0.923 — 0.794	0.916 — 0.710	0.898 — 0.686
	+ Clustering	8	2	0.50	4	6.77	0.990 — 0.972	0.987 — 0.942	0.949 — 0.657	0.972 — 0.848	0.953 — 0.558	0.898 — 0.768	0.974 — 0.815	0.922 — 0.739
				0.25	5	7.09	0.995 — 0.981	0.992 — 0.966	0.974 — 0.794	0.975 — 0.874	0.932 — 0.521	0.939 — 0.825	0.986 — 0.880	0.948 — 0.790
			5	0.50	4	7.90	0.998 — 0.995	0.999 — 0.993	0.989 — 0.914	0.993 — 0.958	0.982 — 0.741	0.924 — 0.839	0.995 — 0.945	0.949 — 0.854
				0.25	5	18.91	1.000 — 0.999	0.999 — 0.999	0.998 — 0.985	0.998 — 0.983	0.987 — 0.869	0.977 — 0.928	0.999 — 0.995	0.984 — 0.936
		16	2	0.50	2	6.24	0.991 — 0.978	0.987 — 0.944	0.944 — 0.673	0.949 — 0.741	0.982 — 0.720	0.931 — 0.811	0.975 — 0.824	0.944 — 0.775
				0.25	2	6.13	0.997 — 0.987	0.991 — 0.965	0.924 — 0.658	0.979 — 0.898	0.995 — 0.942	0.946 — 0.857	0.983 — 0.910	0.957 — 0.852
			5	0.50	2	6.59	1.000 — 0.998	0.998 — 0.988	0.980 — 0.878	0.980 — 0.881	0.991 — 0.864	0.955 — 0.879	0.996 — 0.946	0.968 — 0.882
				0.25	8	9.56	1.000 — 1.000	1.000 — 0.998	0.996 — 0.967	0.999 — 0.996	0.996 — 0.976	0.982 — 0.947	1.000 — 0.995	0.988 — 0.962
		32	2	0.50	6	6.07	0.991 — 0.981	0.990 — 0.948	0.909 — 0.576	0.993 — 0.893	0.888 — 0.470	0.903 — 0.787	0.966 — 0.783	0.917 — 0.731
				0.25	6	6.02	0.997 — 0.994	0.993 — 0.975	0.949 — 0.722	0.993 — 0.947	0.942 — 0.679	0.942 — 0.845	0.984 — 0.889	0.953 — 0.825
			5	0.50	6	6.13	1.000 — 0.999	0.998 — 0.985	0.967 — 0.795	0.997 — 0.956	0.929 — 0.609	0.932 — 0.848	0.991 — 0.890	0.946 — 0.822
				0.25	6	6.69	1.000 — 1.000	0.999 — 0.996	0.994 — 0.961	0.999 — 0.996	0.984 — 0.903	0.975 — 0.925	0.999 — 0.988	0.983 — 0.939
		64	2	0.50	3	5.86	0.993 — 0.982	0.988 — 0.940	0.888 — 0.482	0.979 — 0.825	0.837 — 0.364	0.908 — 0.777	0.958 — 0.751	0.907 — 0.690
				0.25	7	5.83	0.998 — 0.994	0.995 — 0.980	0.949 — 0.740	0.977 — 0.871	0.901 — 0.515	0.931 — 0.827	0.980 — 0.852	0.939 — 0.784
			5	0.50	3	5.73	1.000 — 0.999	0.998 — 0.982	0.949 — 0.685	0.990 — 0.916	0.890 — 0.493	0.927 — 0.828	0.983 — 0.856	0.935 — 0.777
				0.25	7	6.12	1.000 — 1.000	1.000 — 0.999	0.992 — 0.958	0.996 — 0.974	0.951 — 0.731	0.958 — 0.894	0.997 — 0.972	0.969 — 0.897
		128	2	0.50	7	5.83	0.995 — 0.987	0.985 — 0.926	0.897 — 0.469	0.959 — 0.718	0.781 — 0.263	0.905 — 0.763	0.943 — 0.711	0.895 — 0.652
				0.25	2	5.83	0.999 — 0.998	0.995 — 0.969	0.941 — 0.674	0.969 — 0.823	0.850 — 0.407	0.929 — 0.818	0.965 — 0.790	0.927 — 0.742
			5	0.50	7	5.84	0.999 — 0.998	0.996 — 0.970	0.944 — 0.636	0.977 — 0.838	0.846 — 0.381	0.922 — 0.811	0.967 — 0.779	0.922 — 0.731
				0.25	2	5.78	1.000 — 1.000	1.000 — 0.995	0.988 — 0.915	0.992 — 0.943	0.911 — 0.597	0.950 — 0.871	0.990 — 0.917	0.957 — 0.851
	+ Cluster Classifier	64	2	0.50	2	5.86	0.992 — 0.980	0.974 — 0.857	0.977 — 0.924	0.974 — 0.937	0.990 — 0.978	0.967 — 0.930	0.933 — 0.690	0.964 — 0.889
				0.25	7	5.83	0.998 — 0.992	0.988 — 0.921	0.992 — 0.965	0.989 — 0.961	0.996 — 0.989	0.985 — 0.959	0.964 — 0.779	0.983 — 0.931
			5	0.50	2	5.73	1.000 — 1.000	0.993 — 0.953	0.997 — 0.984	0.994 — 0.982	1.000 — 1.000	0.983 — 0.965	0.966 — 0.777	0.984 — 0.943
				0.25	7	6.12	1.000 — 1.000	0.998 — 0.992	0.999 — 0.998	0.999 — 0.996	1.000 — 1.000	0.993 — 0.985	0.993 — 0.935	0.995 — 0.981
GPT-L	No WM	-	-	-	-	4.50	– — –	– — –	– — –	– — –	– — –	– — –	– — –	– — –
	Ours (No Clustering)	-	2	0.50	7	4.51	1.000 — 1.000	0.992 — 0.945	0.835 — 0.377	0.906 — 0.629	0.696 — 0.165	0.932 — 0.787	0.907 — 0.685	0.885 — 0.633
				0.25	4	4.52	1.000 — 1.000	0.996 — 0.981	0.892 — 0.544	0.936 — 0.746	0.732 — 0.264	0.926 — 0.812	0.932 — 0.732	0.898 — 0.694
			5	0.50	7	4.50	1.000 — 1.000	0.995 — 0.968	0.882 — 0.495	0.931 — 0.705	0.719 — 0.224	0.940 — 0.817	0.926 — 0.708	0.902 — 0.678
				0.25	4	4.36	1.000 — 1.000	0.998 — 0.985	0.937 — 0.712	0.964 — 0.849	0.793 — 0.380	0.936 — 0.848	0.961 — 0.812	0.923 — 0.766
	+ Clustering	8	2	0.50	4	5.03	0.988 — 0.969	0.987 — 0.957	0.959 — 0.835	0.966 — 0.863	0.953 — 0.555	0.892 — 0.790	0.977 — 0.905	0.920 — 0.783
				0.25	5	5.65	0.995 — 0.980	0.995 — 0.973	0.981 — 0.898	0.972 — 0.848	0.909 — 0.394	0.941 — 0.838	0.989 — 0.932	0.947 — 0.796
			5	0.50	4	6.24	0.999 — 0.998	0.999 — 0.997	0.996 — 0.983	0.992 — 0.952	0.984 — 0.765	0.918 — 0.851	0.997 — 0.988	0.946 — 0.873
				0.25	5	22.08	1.000 — 1.000	1.000 — 1.000	0.999 — 0.995	0.999 — 0.991	0.977 — 0.773	0.980 — 0.935	0.999 — 0.998	0.984 — 0.931
		16	2	0.50	2	4.58	0.992 — 0.978	0.988 — 0.952	0.948 — 0.780	0.940 — 0.708	0.988 — 0.787	0.930 — 0.830	0.980 — 0.894	0.945 — 0.809
				0.25	2	4.30	0.996 — 0.990	0.993 — 0.973	0.951 — 0.802	0.973 — 0.890	0.999 — 0.983	0.948 — 0.882	0.988 — 0.951	0.962 — 0.892
			5	0.50	2	5.18	1.000 — 1.000	0.999 — 0.994	0.987 — 0.949	0.976 — 0.874	0.994 — 0.899	0.953 — 0.893	0.997 — 0.983	0.968 — 0.906
				0.25	8	9.62	1.000 — 1.000	1.000 — 1.000	0.996 — 0.988	1.000 — 0.999	0.998 — 0.981	0.986 — 0.967	1.000 — 0.999	0.991 — 0.977
		32	2	0.50	6	4.44	0.991 — 0.984	0.993 — 0.979	0.950 — 0.828	0.996 — 0.952	0.877 — 0.477	0.905 — 0.835	0.982 — 0.906	0.922 — 0.805
				0.25	6	4.55	0.999 — 0.994	0.997 — 0.985	0.976 — 0.898	0.996 — 0.962	0.949 — 0.692	0.943 — 0.870	0.994 — 0.962	0.959 — 0.871
			5	0.50	6	4.54	1.000 — 1.000	0.999 — 0.994	0.989 — 0.953	0.998 — 0.985	0.920 — 0.624	0.929 — 0.880	0.999 — 0.978	0.946 — 0.872
				0.25	6	6.15	1.000 — 1.000	1.000 — 0.998	0.998 — 0.990	0.999 — 0.993	0.983 — 0.894	0.974 — 0.940	1.000 — 0.998	0.983 — 0.952
		64	2	0.50	3	4.32	0.995 — 0.988	0.993 — 0.978	0.935 — 0.780	0.984 — 0.890	0.826 — 0.394	0.904 — 0.826	0.983 — 0.896	0.912 — 0.778
				0.25	7	4.34	0.999 — 0.997	0.998 — 0.990	0.966 — 0.893	0.987 — 0.915	0.921 — 0.563	0.927 — 0.857	0.993 — 0.954	0.945 — 0.841
			5	0.50	3	4.36	0.999 — 0.999	0.998 — 0.996	0.978 — 0.909	0.994 — 0.946	0.877 — 0.520	0.921 — 0.860	0.997 — 0.974	0.935 — 0.841
				0.25	7	4.85	1.000 — 1.000	0.999 — 0.999	0.997 — 0.989	0.998 — 0.982	0.962 — 0.730	0.957 — 0.911	0.999 — 0.997	0.972 — 0.913
		128	2	0.50	4	4.34	0.992 — 0.988	0.991 — 0.962	0.921 — 0.712	0.945 — 0.788	0.804 — 0.353	0.895 — 0.807	0.972 — 0.845	0.902 — 0.739
				0.25	4	4.33	1.000 — 0.997	0.997 — 0.984	0.956 — 0.811	0.976 — 0.866	0.866 — 0.459	0.928 — 0.842	0.988 — 0.901	0.936 — 0.797
			5	0.50	4	4.29	1.000 — 1.000	0.999 — 0.994	0.972 — 0.854	0.972 — 0.866	0.851 — 0.459	0.916 — 0.842	0.993 — 0.928	0.930 — 0.807
				0.25	4	4.39	1.000 — 1.000	1.000 — 0.996	0.990 — 0.956	0.994 — 0.958	0.923 — 0.683	0.952 — 0.889	0.999 — 0.985	0.962 — 0.888
	+ Cluster Classifier	64	2	0.50	7	4.32	0.994 — 0.986	0.991 — 0.927	0.983 — 0.963	0.977 — 0.945	0.992 — 0.983	0.968 — 0.937	0.971 — 0.796	0.973 — 0.920
				0.25	7	4.34	0.998 — 0.995	0.994 — 0.958	0.993 — 0.977	0.988 — 0.965	0.998 — 0.992	0.982 — 0.961	0.985 — 0.902	0.985 — 0.954
			5	0.50	3	4.36	1.000 — 1.000	0.996 — 0.979	0.999 — 0.996	0.992 — 0.982	0.999 — 0.999	0.979 — 0.964	0.991 — 0.909	0.985 — 0.963
				0.25	7	4.85	1.000 — 1.000	0.999 — 0.996	0.999 — 0.998	0.998 — 0.997	1.000 — 1.000	0.993 — 0.987	0.999 — 0.989	0.995 — 0.990
RAR-XL	No WM	-	-	-	-	3.13	– — –	– — –	– — –	– — –	– — –	– — –	– — –	– — –
	Ours (No Clustering)	-	2	0.50	3	3.06	0.995 — 0.968	0.963 — 0.745	0.760 — 0.153	0.874 — 0.449	0.697 — 0.105	0.897 — 0.618	0.888 — 0.611	0.855 — 0.498
				0.25	2	3.02	0.999 — 0.989	0.979 — 0.845	0.788 — 0.200	0.907 — 0.551	0.709 — 0.132	0.914 — 0.691	0.907 — 0.661	0.873 — 0.557
			5	0.50	3	3.05	1.000 — 0.995	0.988 — 0.904	0.809 — 0.249	0.925 — 0.634	0.736 — 0.174	0.922 — 0.725	0.916 — 0.693	0.886 — 0.597
				0.25	2	3.14	1.000 — 1.000	0.998 — 0.974	0.898 — 0.466	0.969 — 0.786	0.796 — 0.299	0.942 — 0.812	0.949 — 0.764	0.922 — 0.700
	+ Clustering	8	2	0.50	2	3.43	0.997 — 0.958	0.990 — 0.887	0.931 — 0.447	0.969 — 0.721	0.935 — 0.480	0.955 — 0.734	0.966 — 0.716	0.955 — 0.676
				0.25	2	3.72	0.997 — 0.957	0.994 — 0.917	0.980 — 0.758	0.967 — 0.716	0.948 — 0.626	0.966 — 0.791	0.983 — 0.814	0.971 — 0.771
			5	0.50	8	4.63	1.000 — 0.995	0.998 — 0.975	0.998 — 0.961	0.977 — 0.875	0.950 — 0.705	0.970 — 0.864	0.996 — 0.948	0.975 — 0.868
				0.25	2	9.72	1.000 — 1.000	1.000 — 1.000	1.000 — 0.994	0.999 — 0.980	0.999 — 0.981	0.987 — 0.955	1.000 — 0.993	0.997 — 0.982
		16	2	0.50	1	3.05	0.999 — 0.980	0.990 — 0.900	0.952 — 0.585	0.976 — 0.752	0.916 — 0.411	0.965 — 0.789	0.961 — 0.753	0.961 — 0.718
				0.25	1	3.18	0.999 — 0.986	0.991 — 0.916	0.913 — 0.482	0.977 — 0.789	0.881 — 0.404	0.960 — 0.798	0.971 — 0.802	0.949 — 0.713
			5	0.50	1	3.19	1.000 — 1.000	1.000 — 0.992	0.991 — 0.886	0.995 — 0.944	0.964 — 0.666	0.981 — 0.890	0.988 — 0.890	0.984 — 0.872
				0.25	5	5.80	1.000 — 1.000	1.000 — 0.997	0.999 — 0.982	0.996 — 0.970	0.971 — 0.753	0.973 — 0.901	0.997 — 0.961	0.985 — 0.912
		32	2	0.50	7	3.21	0.998 — 0.982	0.989 — 0.894	0.916 — 0.471	0.938 — 0.664	0.793 — 0.243	0.927 — 0.733	0.951 — 0.732	0.915 — 0.643
				0.25	2	3.08	0.999 — 0.979	0.993 — 0.913	0.941 — 0.478	0.978 — 0.758	0.897 — 0.367	0.936 — 0.730	0.956 — 0.717	0.943 — 0.670
			5	0.50	7	3.31	1.000 — 1.000	0.999 — 0.983	0.975 — 0.769	0.979 — 0.871	0.872 — 0.419	0.951 — 0.838	0.979 — 0.860	0.951 — 0.786
				0.25	2	3.56	1.000 — 1.000	1.000 — 0.997	0.994 — 0.912	0.998 — 0.981	0.967 — 0.719	0.967 — 0.865	0.992 — 0.903	0.978 — 0.872
		64	2	0.50	7	3.05	0.997 — 0.975	0.981 — 0.830	0.815 — 0.215	0.954 — 0.652	0.890 — 0.360	0.910 — 0.692	0.921 — 0.669	0.904 — 0.597
				0.25	3	3.06	0.998 — 0.981	0.989 — 0.911	0.875 — 0.325	0.953 — 0.684	0.834 — 0.276	0.937 — 0.738	0.946 — 0.726	0.924 — 0.636
			5	0.50	7	3.10	1.000 — 0.998	0.996 — 0.966	0.902 — 0.436	0.984 — 0.866	0.936 — 0.514	0.934 — 0.799	0.962 — 0.774	0.940 — 0.728
				0.25	3	3.96	1.000 — 1.000	1.000 — 0.997	0.976 — 0.766	0.993 — 0.948	0.929 — 0.621	0.968 — 0.868	0.987 — 0.893	0.972 — 0.839
		128	2	0.50	2	3.06	0.998 — 0.976	0.979 — 0.800	0.814 — 0.191	0.942 — 0.572	0.793 — 0.163	0.912 — 0.665	0.921 — 0.643	0.892 — 0.546
				0.25	3	3.07	0.998 — 0.988	0.987 — 0.891	0.865 — 0.311	0.952 — 0.659	0.786 — 0.203	0.919 — 0.730	0.933 — 0.699	0.902 — 0.610
			5	0.50	2	3.05	0.999 — 0.996	0.994 — 0.944	0.893 — 0.365	0.972 — 0.775	0.839 — 0.274	0.934 — 0.774	0.948 — 0.729	0.923 — 0.665
				0.25	3	3.39	1.000 — 0.999	0.999 — 0.992	0.964 — 0.718	0.987 — 0.894	0.881 — 0.466	0.947 — 0.845	0.973 — 0.837	0.947 — 0.788
	+ Cluster Classifier	64	2	0.50	8	3.05	1.000 — 1.000	0.993 — 0.954	1.000 — 0.996	0.997 — 0.977	1.000 — 0.997	0.992 — 0.957	0.937 — 0.725	0.986 — 0.931
				0.25	1	3.06	1.000 — 0.998	0.996 — 0.973	0.999 — 0.994	0.997 — 0.979	1.000 — 0.995	0.994 — 0.967	0.961 — 0.779	0.990 — 0.946
			5	0.50	6	3.10	1.000 — 1.000	0.998 — 0.988	1.000 — 0.999	1.000 — 0.997	1.000 — 0.999	0.997 — 0.982	0.971 — 0.794	0.994 — 0.961
				0.25	1	3.96	1.000 — 1.000	1.000 — 0.997	1.000 — 1.000	1.000 — 0.999	1.000 — 1.000	0.998 — 0.991	0.991 — 0.923	0.998 — 0.984

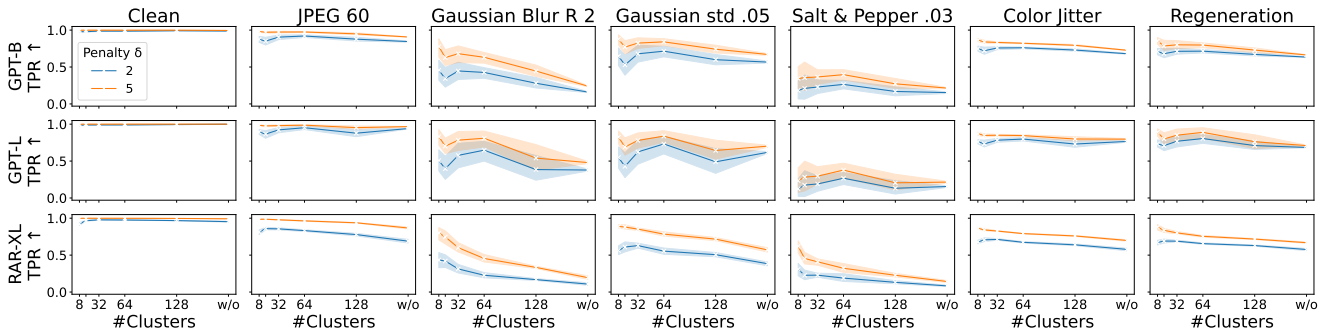
Table 7. **Full experimental results on *Perturbation Set A*** in terms of **AUC — TPR@FPR=1% after prefix tuning**. The reported results use the best performing prefix (from 8 tested prefixes for each setting) in each group where the group is defined by the model, method, number of clusters k, penalty δ and green token fraction γ. Best prefixes were chosen by taking those with best average TPR@FPR=1%, averaged across all perturbations (columns).

	+xy=(6,7)	crop=0.11	crop=0.33	hflip	prsp=0.3	rot=33	rot=7	scl=0.9	scl=1.1	scl=1.5	tl=0.7
LlamaGen GPT-B 256×256											
No Clusters	0.010	0.145	0.085	0.013	0.015	0.013	0.025	0.022	0.071	0.001	0.989
+ Token Classifier	0.017	0.094	0.129	0.011	0.012	0.013	0.020	0.025	0.088	0.001	1.000
64 Clusters	0.089	0.296	0.233	0.067	0.017	0.014	0.118	0.148	0.650	0.011	1.000
+ Token Classifier	0.051	0.248	0.304	0.043	0.011	0.007	0.115	0.132	0.665	0.019	1.000
+ Cluster Classifier	0.064	0.332	0.276	0.059	0.009	0.003	0.143	0.141	0.593	0.025	0.999
LlamaGen GPT-L 384×384											
No Clusters	0.013	0.135	0.151	0.020	0.017	0.011	0.042	0.049	0.224	0.004	0.999
+ Token Classifier	0.009	0.136	0.206	0.011	0.042	0.003	0.035	0.050	0.201	0.001	0.999
64 Clusters	0.183	0.378	0.403	0.154	0.033	0.015	0.561	0.435	0.917	0.060	0.999
+ Token Classifier	0.122	0.358	0.395	0.111	0.009	0.005	0.530	0.431	0.911	0.072	1.000
+ Cluster Classifier	0.143	0.386	0.459	0.117	0.012	0.004	0.529	0.422	0.844	0.099	1.000
RAR-XL 256×256											
No Clusters	0.035	0.221	0.216	0.024	0.025	0.013	0.069	0.111	0.576	0.003	0.987
+ Token Classifier	0.025	0.406	0.498	0.021	0.064	0.011	0.205	0.336	0.972	0.003	1.000
64 Clusters	0.062	0.405	0.404	0.033	0.026	0.012	0.177	0.312	0.838	0.009	0.996
+ Token Classifier	0.059	0.533	0.615	0.043	0.077	0.008	0.495	0.616	0.995	0.013	1.000
+ Cluster Classifier	0.038	0.465	0.526	0.040	0.051	0.005	0.365	0.547	0.989	0.011	1.000

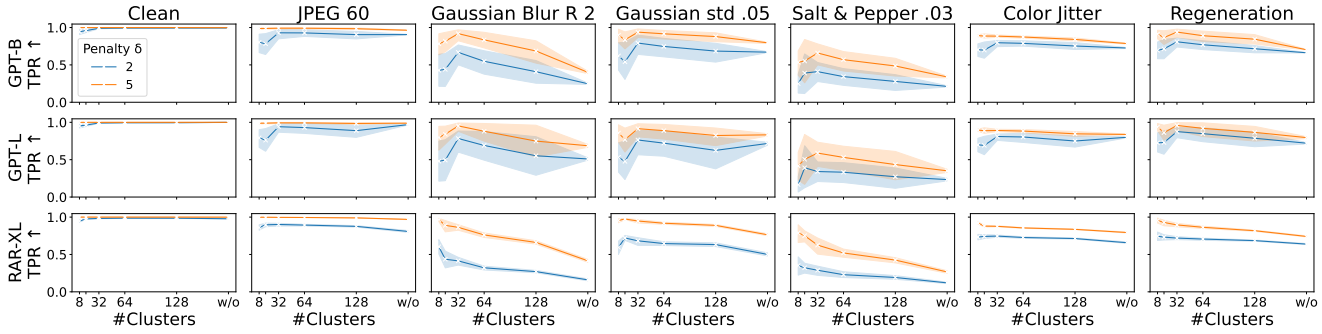
Table 8. With geometric transformations: Detection accuracy (TPR@FPR=1%) with our proposed methods (fixed $\gamma = 0.25$ and $\delta = 5$), deployed with LlamaGen (GPT-B and GPT-L) and RAR-XL. The geometric transformations are: (1) +xy: translation, (2) crop by a factor, effectively lowering resolution, (3) horizontal flip, (4) prsp: perspective transform, (5) rot: rotation by a degree, (6) scl: scaling by a factor while cropping if the image is larger than original, (7) crop_tl: cropping the top-left part of the image by a factor.

	rot=33	rot=7	scl=0.5	scl=0.7	scl=0.9	scl=1.1	scl=1.5
LlamaGen GPT-B 256×256							
No Clusters	0.988	0.989	0.884	0.988	0.994	0.995	0.995
+ Token Classifier	1.000	1.000	0.997	0.999	1.000	1.000	1.000
64 Clusters	1.000	1.000	0.997	1.000	1.000	1.000	1.000
+ Token Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000
+ Cluster Classifier	0.999	1.000	1.000	1.000	1.000	1.000	1.000
LlamaGen GPT-L 384×384							
No Clusters	1.000	1.000	0.982	0.998	0.999	1.000	1.000
+ Token Classifier	1.000	1.000	0.999	1.000	1.000	1.000	1.000
64 Clusters	1.000	1.000	1.000	1.000	1.000	1.000	1.000
+ Token Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000
+ Cluster Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000
RAR-XL 256×256							
No Clusters	0.994	0.990	0.956	0.998	0.999	1.000	1.000
+ Token Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000
64 Clusters	0.998	0.998	0.993	1.000	1.000	1.000	1.000
+ Token Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000
+ Cluster Classifier	1.000	1.000	1.000	1.000	1.000	1.000	1.000

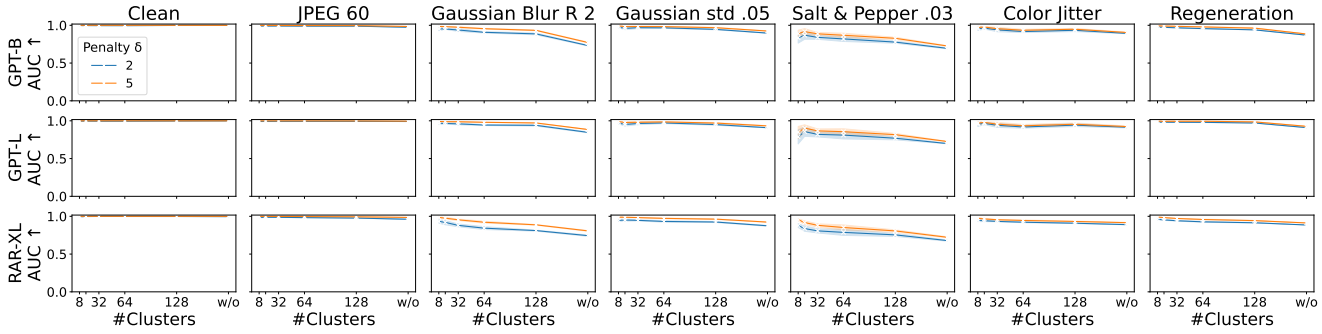
Table 9. With reverted geometric transformations: Detection accuracy (TPR@FPR=1%) with our proposed methods (fixed $\gamma = 0.25$ and $\delta = 5$), deployed with LlamaGen (GPT-B and GPT-L) and RAR-XL.



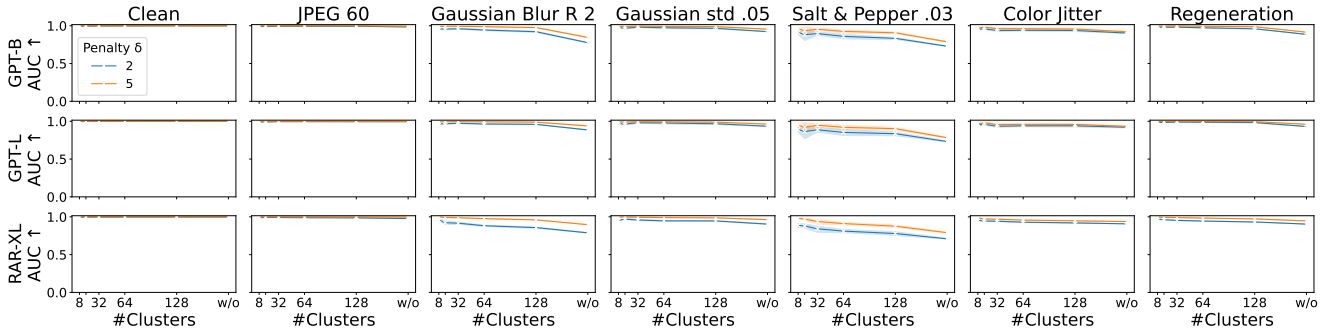
(a) TPR@FPR=1% for green token fraction $\gamma = 0.5$



(b) TPR@FPR=1% for green token fraction $\gamma = 0.25$



(c) AUC for green token fraction $\gamma = 0.5$



(d) AUC for green token fraction $\gamma = 0.25$

Figure 9. **Ablation of detection characteristics** across models, perturbations (Perturbation Set A), amount of clusters, penalty δ, and green token fraction γ settings. The results are reported over multiple prefixes (8 for RAR-XL as well as LlamaGen GPT-B and GPT-L), where the standard deviation σ is indicated by the colored areas. The lines are averaged over prefixes. Note that the cluster classifier is not used here.

	hflip	prsp=0.3	rot=33	rot=7	scl=0.7	scl=1.5	scl=1.5 + gaussian blur R=3	SyncSeal
LlamaGen GPT-B 256×256								
No Clusters	0.893	0.927	0.917	0.943	0.703	0.966	0.358	0.989
+ Token Classifier	0.947	0.986	0.977	0.985	0.919	0.992	0.858	0.996
64 Clusters	0.991	0.997	0.990	0.997	0.985	0.997	0.936	0.999
+ Token Classifier	0.997	1.000	0.998	0.999	0.998	0.999	0.997	1.000
+ Cluster Classifier	0.991	0.997	0.984	0.997	0.996	0.997	0.982	1.000
LlamaGen GPT-L 384×384								
No Clusters	0.849	0.916	0.970	0.988	0.553	0.988	0.630	0.999
+ Token Classifier	0.888	0.932	0.986	0.994	0.725	0.996	0.953	1.000
64 Clusters	0.989	0.994	0.999	1.000	0.933	1.000	0.990	1.000
+ Token Classifier	0.992	0.996	1.000	1.000	0.951	1.000	1.000	1.000
+ Cluster Classifier	0.987	0.992	0.995	1.000	0.932	1.000	0.996	1.000
RAR-XL 256×256								
No Clusters	0.912	0.959	0.841	0.921	0.858	0.979	0.376	0.994
+ Token Classifier	0.995	0.998	0.993	0.997	0.992	0.997	0.998	0.999
64 Clusters	0.971	0.987	0.937	0.983	0.945	0.993	0.661	1.000
+ Token Classifier	1.000	1.000	0.996	0.999	0.997	1.000	1.000	1.000
+ Cluster Classifier	0.999	1.000	0.991	0.999	0.995	1.000	0.999	1.000

Table 10. With SyncSeal-reverted geometric transformations: Detection accuracy (TPR@FPR=1%) with our proposed methods (fixed $\gamma = 0.25$ and $\delta = 5$), deployed with LlamaGen (GPT-B and GPT-L) and RAR-XL.

Figure 10. Visual examples of unwatermarked and watermarked images generated with **LlamaGen (GPT-B)** across different settings.

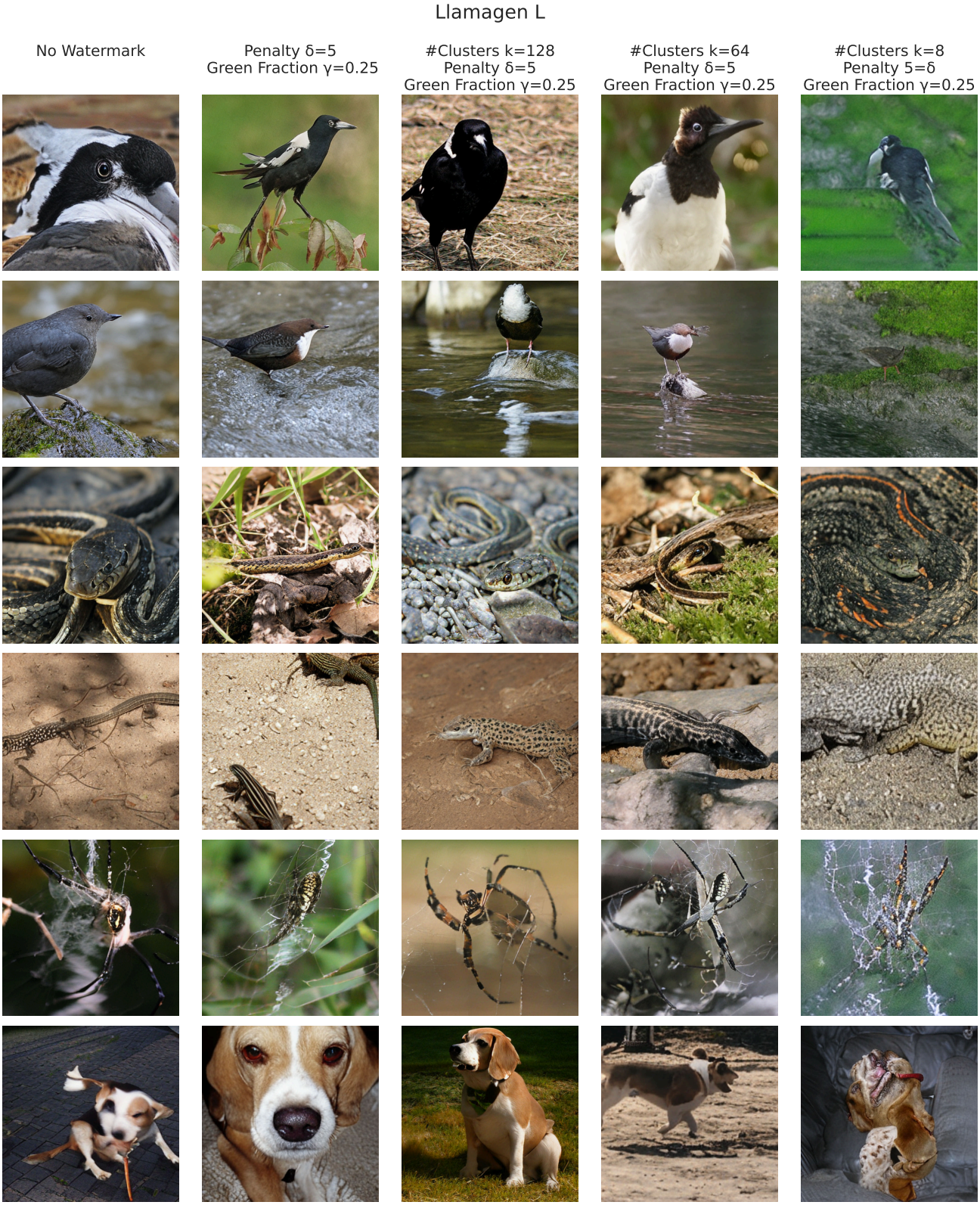


Figure 11. Visual examples of unwatermarked and watermarked images generated with **LlamaGen (GPT-L)** across different settings.

Figure 12. Visual examples of unwatermarked and watermarked images generated with **RAR-XL** across different settings.