

IMAGE RESTORATION UNDER FOG, DARKNESS, AND MOTION BLUR FOR LICENSE PLATE RECOGNITION

Yusuf Morsi, Girish Krishnan, and Philip Pincencia

Department of Electrical and Computer Engineering, University of California San Diego

ABSTRACT

Suboptimal visual conditions such as fog, low light, and motion blur are not only quite common in the world, but significantly decrease the quality of captured images, hindering their usability for real-world applications such as automatic license-plate recognition systems. Previous research in digital image restoration has equipped the restorer with dark-channel de-hazing, gamma correction, and Wiener filtering, but most studies are reliant on synthetic/benchmark datasets, and are lacking in natural dataset evaluation. With the goal of addressing this gap, we have collected a real-world dataset of scenes under foggy, night, motion-blur, and clear conditions in order to analyze how various image-processing algorithms improve visual quality and feature clarity to aid in license-plate detection and text readability. We apply and tune defogging, undarkening, and deblurring algorithms for each category, and compare their effectiveness using metrics such as PSNR, SSIM, NIQE, and BRISQUE. We hope to show the strengths and weaknesses of classical enhancement methods under varying conditions, and quantify their impact on the accuracy of license-plate localization and optical character recognition (OCR).

Index Terms— Image enhancement, defogging, deblurring, low-light correction, license plate recognition

1. INTRODUCTION

Common real-world conditions such as fog, low light, and motion blur are common in everyday photography, making it difficult to capture clear images. Cameras/CCTVs in uncontrolled environments; such as traffic intersections, parking lots, or highways; record images and videos with low contrast, faded edges, and rough lighting. These distortions restrict how well basic vision systems read and analyze what is in front of them, especially when it comes to cases as practical as license-plate detection or traffic monitoring.

For fog, scattered light and washed out colors make it difficult to see scene night or low-light. In the case of night/low-light conditions, sensor noise and poor exposure make smaller details difficult to see. In the event the camera/photographer moves quickly (especially common if in a moving vehicle), motion blur stretches objects and removes sharp texture, mak-

ing any text present in them impossible to read. These effects work in their own way to hinder the visibility of the scene, and makes images and their text harder to detect.

In this time of AI hype, our project's main focus is to return to the basics and investigate how classical digital image processing methods can mitigate these issues and improve the clarity and readability of real images. We are collecting a small custom dataset of visible license plates, that includes foggy, low-light, motion-blur, and clear-day photos collected around different parts of the United States, including San Diego, San Francisco, Phoenix, San Jose, Los Angeles, Chicago, and Pittsburgh with phones and dashcams. After testing various classical digital image processing algorithms such as defogging, undarkening, and deblurring, we will compare how much each method improves overall image quality and the readability of license-plate text in normal outdoor conditions.

1.1. Contributions and Innovation

Our project goal is to evaluate how much traditional image processing can continue to help in a field that is at this time, dominated by deep learning. From scratch, we are collecting a custom small real-world dataset of cars and license plates in fog, low-light, and motion-blur conditions across several U.S. cities, before benchmarking simple restoration pipelines against more modern ones.

The key contributions are:

- Collecting a manually captured dataset that covers four common real-life conditions (daylight, fog, darkness, and motion blur)
- Developing a unified evaluation pipeline that applies enhancement to a public ALPR model (Fast-ALPR) in order to measure how much visibility restoration impacts detection and OCR accuracy.
- For each type, we compare at least two algorithms, one of which being classical (e.g., Dark Channel Prior, Richardson–Lucy, Adaptive Gamma), to note trade-offs between the two.
- We will implement a condition-detection pipeline that automatically routes each image to the most appropriate

ate enhancement step, so that we can have cascading of multiple corrections when necessary.

Rather than building a new model, we will show how the tuning of existing digital processing algorithms can make off-the-shelf recognition systems noticeably more reliable in real environments.

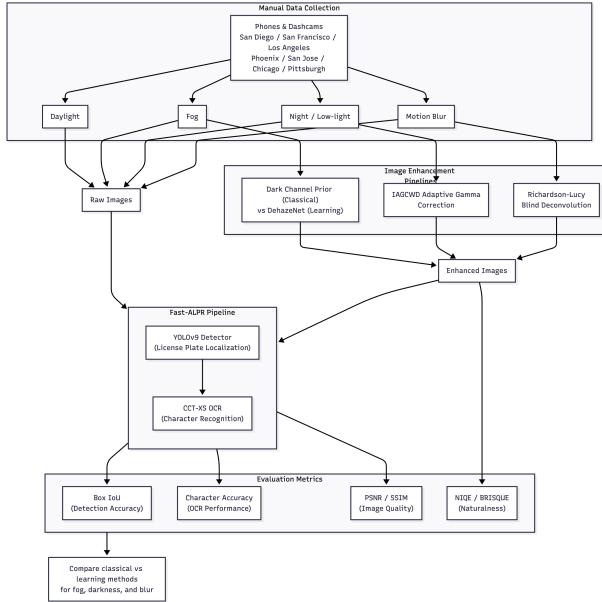


Fig. 1: Overall project workflow showing main steps: image collection, digital image processing (defogging, undarkening, deblurring), and evaluation using Fast-ALPR.

2. RELATED WORKS

Prior works can be broadly classified into learning-based restoration models for specific degradations such as low light, haze, or motion blur, and classical image processing methods designed to address these distortions through physically inspired priors and signal models. In this section, we review both categories and highlight their strengths, limitations, and relevance to our project.

2.1. Low-Light Enhancement and Nighttime Recognition

Low-light imaging severely affects license plate readability by reducing contrast, amplifying sensor noise, and suppressing fine details. Saputra *et al.* [1] addressed this problem through the URetinex-Net model, which unfolds a Retinex-based optimization into a trainable network that separates illumination and reflectance layers. Their pipeline combined URetinex-Net with a TRBA (Transformer–Bidirectional LSTM–Attention) OCR recognizer, achieving an 80.11% recognition rate in night scenes, corresponding to a 5.11% improvement over baseline methods without enhancement.

The network excels at decomposing illumination and preserving text texture; however, it remains limited by synthetic training data and may generalize poorly to unseen lighting distributions. Moreover, its reliance on global illumination priors makes it computationally demanding for real-time use in embedded systems.

Complementing this, the classical Improved Adaptive Gamma Correction with Weighted Distribution (IAGCWD) algorithm [2] provides a non-learning alternative that adaptively brightens underexposed regions while maintaining local contrast. It modifies the histogram-based gamma mapping using weighted probability distributions and CDF truncation, preventing over-amplification in already bright regions. While it cannot learn semantic priors like deep networks, its interpretability, parameter efficiency, and robustness make it highly suitable for low-power applications or datasets lacking paired supervision. The key limitation here is its dependence on global statistics, which may inadequately handle spatially varying illumination.

2.2. Haze and Fog Removal

He *et al.* [3] proposed the seminal Dark Channel Prior (DCP) method, which estimates transmission maps from the minimum intensity of local patches under the assumption that most outdoor scenes contain at least one dark pixel in each color channel. The de-hazing process reconstructs the latent scene radiance by inverting the Koschmieder model. DCP remains one of the most widely adopted priors due to its mathematical simplicity and ability to restore global contrast; however, its pixel-wise assumptions can over-saturate bright areas and amplify noise in sky-dominated regions. Subsequent deep-learning approaches often embed DCP as a regularization term but still inherit this limitation under heavy haze or non-uniform illumination.

Even though the method is simple, DCP continues to outperform many black-box deep models in visibility restoration for small datasets. For license plate recognition, the method effectively enhances edge contrast and color balance, though it lacks adaptive noise suppression and may introduce halo artifacts at sharp boundaries.

2.3. Motion Deblurring and Dynamic Scene Restoration

Gong *et al.* [4] introduced the LP-Blur dataset (which is the first large-scale, real-captured dataset of paired blurred and sharp license plate images) along with LPDGAN, a generative adversarial deblurring network designed for text preservation. The model integrates multi-scale latent fusion, text reconstruction through OCR feature consistency, and a partition discriminator that enforces local sharpness at the character level. LPDGAN achieved a PSNR of 29.95 dB under normal light and 30.96 dB in low light, outperforming previous methods by over 1 dB and improving OCR accuracy

by 21.24%. Its primary strength is its character-aware adversarial training, which restores semantic legibility rather than mere pixel sharpness. Nevertheless, the architecture’s multi-scale Transformer components result in heavy computational cost and it may not be necessarily effective on other alphabets or plate styles.

Another similar line of work by Na *et al.* [5] proposed MF-LPR², which uses multi-frame information through optical flow alignment. Using FlowFormer++ for inter-frame motion estimation, the method aggregates frames using temporal filtering and spatial refinement modules to suppress inconsistent motion vectors. On their RLPR dataset, MF-LPR² achieved an OCR accuracy of 86.44%, outperforming eleven state-of-the-art baselines, and significantly improving both PSNR (16.35 dB) and SSIM (0.35). The method’s interpretability makes it good for forensic applications. However, it requires dense optical flow computation, making it unsuitable for real-time traffic systems. Its performance also deteriorates when optical flow estimation fails due to occlusion or severe blur.

Classical (non-learning based) deblurring methods such as Richardson–Lucy (RL) deconvolution [6, 7] are still important due to their strong physical grounding (as the methods are interpretable). The RL algorithm iteratively maximizes the Poisson likelihood of the observed image, assuming a known blur kernel. When combined with Gaussian smoothing and appropriate boundary handling [8, 9], RL can recover sharp details. Modern blind deblurring variants [10, 11, 12, 13] have introduced kernel estimation schemes based on motion trajectory priors to give further improvements to performance. Even though deep models outperform RL in visual fidelity, the RL algorithm is better for controlled datasets where physical blur models can be reasonably approximated. Its main weaknesses, however, are noise amplification and computational instability (especially when the blur kernel is incorrectly estimated).

2.4. Super-Resolution and Benchmark Datasets

Recent efforts have also focused on super-resolution (SR) and making suitable datasets for degraded license plate images that need restoration. For example, Nascimento *et al.* [14] proposed the UFPR-SR-Plates dataset, which provides 100,000 paired images captured under different environmental conditions. They benchmarked several SR models including Real-ESRGAN, SR3, LPSRGAN, PLNET, and LCDNet using a GP-LPR OCR backend. The study showed that combining multiple restored frames via a majority-vote-by-character scheme increased full-plate recognition accuracy from 2.2% for raw low-resolution inputs to 42.3%, while the best individual model (LCDNet) achieved 61.4% accuracy at high resolution. The dataset’s strength is its scale and real-world diversity. However, it currently lacks night-time and cross-country data, which our project contributes.

Diffusion-based SR models such as SR3 provide good quality but remain computationally expensive for real-time deployment.

2.5. Current Research Trends and Gaps

The current trend is to combine learning-based methods with classical methods that provide interpretable (and physically meaningful) priors. Diffusion and Transformer-based architectures such as LP-Diff [15] combine diffusion denoising with fine-grained plate structure constraints, but such models demand extensive training data and are less suited to low-data, real-time contexts. Classical methods like RL deconvolution, the dark channel prior (DCP), and adaptive gamma correction remain competitive due to their interpretability, tunable behavior, and independence from learned priors. Their low computational cost makes them ideal for systematic evaluation on real-world data. Our project will thus focus on the use of these classical image processing algorithms as pre-processing methods before feeding the image into deep learning models for license plate detection and recognition.

3. PROPOSED WORK

3.1. Models for License Plate Detection and Recognition

The Fast-ALPR pipeline [16] involves two deep-learning components trained separately: a YOLOv9-based license plate detector and a Compact Convolutional Transformer (CCT)-based OCR recognizer. Figure 2 shows the process of how Fast-ALPR works, beginning with the image enhancement stage, and leading to the license plate localization and text decoding stage.

3.1.1. YOLOv9 for License Plate Detection

The YOLOv9 detector [17] builds on the one-stage object detection family with a *Programmable Gradient Information Bottleneck (C2f-PGB)* and a *Generalized Efficient Layer Aggregation Network (GELAN)* backbone (Figure 3). For an input image $I \in \mathbb{R}^{H \times W \times 3}$, the detector predicts K bounding boxes $\{b_k\}_{k=1}^K$ and class confidence scores $\{s_k\}_{k=1}^K$ such that

$$b_k = (x_k, y_k, w_k, h_k), \quad s_k = \sigma(\mathbf{W}_h^\top f(I)), \quad (1)$$

where $f(\cdot)$ represents the shared convolutional backbone, and σ is the logistic activation. During training, YOLOv9 minimizes a compound loss

$$\mathcal{L}_{\text{det}} = \lambda_{\text{box}} \mathcal{L}_{\text{ClIoU}} + \lambda_{\text{cls}} \mathcal{L}_{\text{BCE}} + \lambda_{\text{obj}} \mathcal{L}_{\text{conf}}, \quad (2)$$

where $\mathcal{L}_{\text{ClIoU}}$ is the Complete IoU loss for the regression of the bounding box, \mathcal{L}_{BCE} represents binary cross-entropy terms for class prediction and $\mathcal{L}_{\text{conf}}$ represents the confidence loss, a binary cross-entropy term that penalizes incorrect predictions of whether a bounding box truly contains a license plate or whether it is just the background.

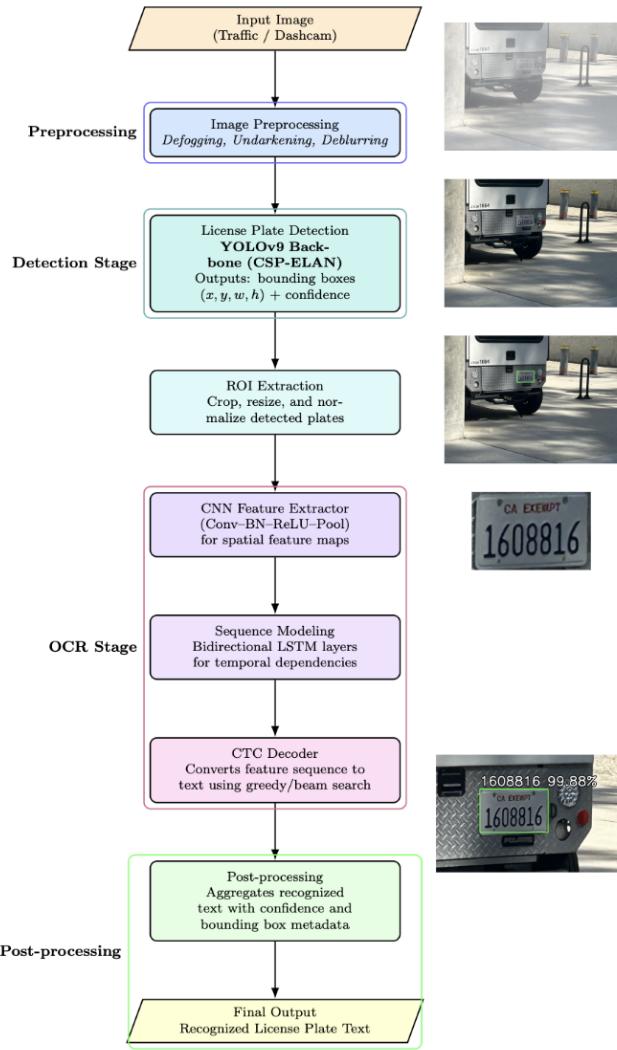


Fig. 2: Architecture of Fast-ALPR. The input image is first enhanced (defogging, underdarkening, deblurring) before undergoing detection with a YOLOv9-based license plate detector. Potential regions of interest that were detected are processed by a CNN OCR network for character recognition, and final text results are post-processed with bounding boxes.

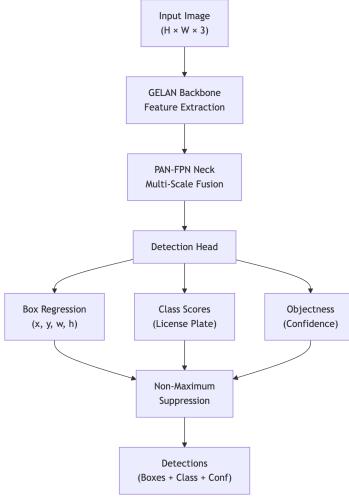


Fig. 3: YOLOv9 architecture showing the GELAN backbone, PAN-FPN neck, and three-branch detection head outputting bounding boxes, class scores, and confidence.

3.1.2. CCT-XS OCR for Character Recognition

For character-level optical recognition, the CCT model combines convolutional token embedding with lightweight Transformer encoders and a Bidirectional Long Short-Term Memory (BiLSTM) sequence head trained with Connectionist Temporal Classification (CTC) loss. Given a cropped license plate image P , the OCR model first extracts visual features with convolutional layers and then processes them as a sequence through the Transformer and BiLSTM modules to capture spatial and contextual information. The output sequence is decoded by a CTC layer to produce the final text prediction \hat{T} , bringing out the final recognized license plate number.

3.2. Blind Motion Deblurring

We will use a classical Richardson–Lucy (RL) deconvolution framework [6, 7], extended with a grid-based estimation of the motion blur Point-Spread Function (PSF).

3.2.1. Image model and RL deconvolution

The blurred image B is the convolution of an unknown latent sharp image I with a blur kernel k and additive noise η :

$$B = I * k + \eta, \quad (3)$$

where $*$ represents 2D convolution. Given an estimate of k , RL refines the estimate (iteratively) \hat{I} by maximizing the Poisson likelihood of B conditioned on $\hat{I} * k$:

$$\hat{I}^{(t+1)} = \hat{I}^{(t)} \cdot \left[\frac{B}{\hat{I}^{(t)} * k + \epsilon} * k^* \right], \quad (4)$$

where k^* is k flipped horizontally and vertically, and $\epsilon \ll 1$ prevents division by zero.

However, unregularized RL tends to amplify noise and overemphasize high frequencies after multiple iterations [18]. To suppress ringing and stabilize convergence, we incorporate Gaussian smoothing after each iteration:

$$\hat{I}^{(t+1)} \leftarrow G_\sigma * \hat{I}^{(t+1)}, \quad (5)$$

where G_σ is a Gaussian filter with standard deviation σ .

3.2.2. PSF estimation

The accuracy of non-blind deconvolution fundamentally depends on the estimated PSF [10]. In the blind setting, where k is unknown, jointly estimating I and k is severely ill-posed since both contribute multiplicatively in the Fourier domain, leading to many degenerate solutions. Moreover, optimization-based blind deblurring often converges to trivial or overly smooth kernels unless carefully initialized [13].

To avoid such instability, we parameterize k as a *linear motion* PSF characterized by its length L and orientation θ :

$$k(L, \theta)_{i,j} = \frac{1}{Z} \cdot \delta(j - i \tan(\theta)), \quad i, j \in [-L/2, L/2], \quad (6)$$

where $Z = \sum_{i,j} k_{i,j}$ normalizes the kernel. This formulation captures the dominant form of motion blur caused by camera or object translation during exposure [11]. By constraining k to a physically plausible parameter space, we avoid the over-parameterization that often destabilizes free-form kernel estimation [12].

Rather than solving a fragile continuous optimization over (L, θ) , we discretize the parameter space into a uniform grid:

$$\mathcal{K} = \{ k(L, \theta) \mid L \in [L_{\min}, L_{\max}], \theta \in [0, 360^\circ] \}. \quad (7)$$

Each kernel $k_i \in \mathcal{K}$ is applied to a downsampled version of B , followed by a few RL iterations (T_{eval}) to obtain a preliminary estimate \hat{I}_i . This produces a set of candidate reconstructions representing distinct motion hypotheses.

To quantify the plausibility of each candidate, we compute a sharpness score S_i based on the mean variance of the Laplacian across color channels:

$$S_i = \frac{1}{3} \sum_{c \in \{R, G, B\}} \text{Var}[\nabla^2 \hat{I}_{i,c}], \quad (8)$$

where ∇^2 denotes the Laplacian operator. This metric reflects the intuitive notion that well-deblurred images exhibit sharper edges and stronger local contrast while minimizing residual blur [19]. The PSF corresponding to the maximum sharpness is then selected:

$$k^* = \arg \max_{k_i \in \mathcal{K}} S_i. \quad (9)$$



Fig. 4: An original motion-blurred image (top-left) and the result after applying Blind Motion Deblurring (top-right). The bottom-left and bottom-right images show the results of Fast-ALPR on the blurred and deblurred images, respectively.

3.2.3. Boundary stabilization

The ringing issues mentioned previously are mitigated slightly using Eq. (5), which applies Gaussian smoothing throughout the image. However, a more severe form of ringing, known as Gibbs oscillation, tends to occur near image borders due to discontinuities introduced by finite-support convolution and the implicit assumption of periodic boundaries. To prevent this, the blurred input is extended by p pixels using border replication:

$$B_{\text{ext}} = \text{Replicate}(B, p), \quad (10)$$

which ensures continuity across the convolution boundaries [8, 9]. After deconvolution, the padding region is removed to obtain the final deblurred image:

$$\hat{I} = \text{Crop}(\hat{I}_{\text{ext}}, p).$$

The full deblurring pipeline is summarized in Algorithm 1. In short, the method evaluates a discrete set of motion kernels, reconstructs provisional images via RL updates, and selects the kernel that produces the sharpest and most stable restoration. Figure 4 shows this algorithm being used on an example motion-blurred image, and a comparison of the results of running Fast-ALPR on the original and de-blurred image.

3.2.4. Comparator: Wiener Filter for Motion Deblurring

We are also planning on implementing a Wiener filter for motion deblurring algorithm, based on the public implementation of [20]. This approach carries out deconvolution in the frequency domain, with a closed-form solution where we have both inverse filtering and noise suppression.

Algorithm 1 Grid-Based Blind Motion Deblurring

Require: Blurred image B , PSF grid \mathcal{K} , evaluation iterations T_{eval} , final iterations T_{final} , border extension p

Ensure: Restored image \hat{I}

- 1: $B_{\text{ext}} \leftarrow \text{Replicate}(B, p)$
- 2: $\mathcal{S} \leftarrow \emptyset$
- 3: **for all** $k_i \in \mathcal{K}$ **do**
- 4: Initialize $\hat{I}_i^{(0)} \leftarrow B_{\text{ext}}$
- 5: **for** $t = 1 \dots T_{\text{eval}}$ **do**
- 6: $r \leftarrow \text{clip}\left(\frac{B_{\text{ext}}}{\hat{I}_i^{(t-1)} * k_i + \epsilon}, r_{\min}, r_{\max}\right)$
- 7: $\hat{I}_i^{(t)} \leftarrow \hat{I}_i^{(t-1)} \cdot (r * k_i^*)$
- 8: $\hat{I}_i^{(t)} \leftarrow G_\sigma * \hat{I}_i^{(t)}$
- 9: **end for**
- 10: $S_i \leftarrow \text{Var}[\nabla^2 \hat{I}_i^{(T_{\text{eval}})}]$
- 11: $\mathcal{S} \leftarrow \mathcal{S} \cup \{(k_i, S_i)\}$
- 12: **end for**
- 13: $k^* \leftarrow \arg \max_{(k_i, S_i) \in \mathcal{S}} S_i$
- 14: Run RL deconvolution on B_{ext} using k^* for T_{final} iterations
- 15: $\hat{I} \leftarrow \text{Crop}(\hat{I}_{\text{final}}, p)$
- 16: **return** \hat{I}

It uses the same image model, but now we are using the Wiener filter, which is just a naive inverse filter with a regularization term proportional to the noise-to-signal ratio (NSR):

$$\hat{I}(u, v) = \frac{\mathcal{F}\{k\}^*(u, v)}{|\mathcal{F}\{k\}(u, v)|^2 + \text{NSR}} \mathcal{F}\{B\}(u, v), \quad (11)$$

where $\mathcal{F}\{k\}^*$ is the complex conjugate of the blur spectrum. Then, the restored image is obtained by inverse Fourier transform:

$$\hat{I} = \mathcal{F}^{-1}\{\hat{I}(u, v)\}.$$

Just before the frequency-domain filtering, we applied a smooth windowing function that attenuates image borders to prevent ringing artifacts:

$$B_{\text{tapered}}(x, y) = B(x, y) w_x(x) w_y(y), \quad (12)$$

where $w_x(x) = \frac{1}{2} [\tanh((x+\gamma/2)/\beta) - \tanh((x-\gamma/2)/\beta)]$ and similarly for $w_y(y)$. This ensures continuity near the image edges.

3.3. Adaptive Gamma Correction for Low-Light Enhancement

For underexposed or low-light images in our dataset, we will use the *Improved Adaptive Gamma Correction with Weighted Distribution (IAGCWD)* algorithm created by Cao *et al.* [2]. This technique extends classical adaptive gamma correction by introducing two mechanisms: (1) a *negative-image transform* to enhance globally bright images, and (2) a *CDF truncation scheme* to prevent over-enhancement in dimmed regions. The method modulates pixel-wise gamma parameters

in an adaptive manner based on the statistical structure of the input image histogram, producing consistent brightness correction.

3.3.1. Mathematical Formulation

Given a grayscale luminance channel $I(x, y) \in [0, 255]$, the pixel transformation is defined as

$$T(l) = \text{round}\left(l_{\max} \left(\frac{l}{l_{\max}}\right)^{\gamma(l)}\right), \quad (13)$$

where $l_{\max} = 255$ and $\gamma(l)$ is the intensity-dependent gamma parameter derived from the cumulative distribution function (CDF) of the gray-level histogram $p(l)$:

$$\gamma(l) = 1 - c(l), \quad c(l) = \sum_{x=0}^l p(x). \quad (14)$$

The CDF controls local tone mapping, brightening darker pixels with smaller $\gamma(l)$ while preserving structure in well-exposed regions. To improve the adaptivity of $p(l)$, a *weighted distribution* $p_w(l)$ is constructed as

$$p_w(l) = \frac{p(l) - p_{\min}}{p_{\max} - p_{\min}} p_{\max}^\alpha, \quad (15)$$

where $\alpha \in (0, 1]$ controls the flattening strength of the histogram; smaller α reduces dominance of peak intensities. The normalized weighted distribution is used to recompute a smooth CDF $c_w(l)$, which stabilizes the mapping under nonuniform illumination.

For bright images, IAGCWD applies the same transformation to the *negative image* $I'(x, y) = 255 - I(x, y)$, which gives $T'(I')$, and then inverts it:

$$I_e(x, y) = 255 - T'(I'(x, y)). \quad (16)$$

This ensures that large high-intensity regions (saturated whites) are effectively treated as low-intensity regions during correction.

To avoid over-enhancement and loss of detail in bright regions of dimmed images, Cao *et al.* introduce a *CDF truncation scheme*:

$$\gamma'(l) = \max(\tau, 1 - c_w(l)), \quad (17)$$

where $\tau \in [0, 1]$ is a lower bound on the gamma value. This prevents $\gamma'(l)$ from approaching zero, thus constraining the brightness expansion of high-intensity pixels. The overall algorithm is summarized in Algorithm 2.

3.3.2. Parameter Tuning

The algorithm involves four tunable parameters:

- α : histogram weighting coefficient controlling contrast flattening.
- τ : truncation parameter controlling enhancement strength.
- T_t : expected global average intensity (typically $T_t = 112$ for 8-bit images).
- τ_t : threshold for classifying bright vs. dimmed images (empirically $\tau_t = 0.3$).

According to the paper, $\alpha = 0.25$ and $\tau = 0.5$ give effective enhancement for bright scenes, while $\alpha = 0.75$ and $\tau = 0.5$ work well for dimmed ones. Figure 5 shows an example dark image alongside its processed image using these parameters. However, we plan to conduct an ablation study where these parameters are tuned further to explore the resulting performance on the license plate detection and OCR task.

Algorithm 2 Improved Adaptive Gamma Correction with Weighted Distribution (IAGCWD)

Require: Input luminance image I , parameters $\alpha, \tau, T_t, \tau_t$
Ensure: Enhanced image I_e

- 1: Compute global mean brightness $m = \frac{1}{MN} \sum_{x,y} I(x, y)$
- 2: Determine image type: $t = (m - T_t)/T_t$
- 3: **if** $t > \tau_t$ **then** ▷ Bright image
- 4: Compute negative image $I' = 255 - I$
- 5: Enhance I' via weighted CDF and Eq. (15)–(17) to obtain I'_e
- 6: $I_e = 255 - I'_e$
- 7: **else if** $t < -\tau_t$ **then** ▷ Dimmed image
- 8: Compute weighted histogram $p_w(l)$ using Eq. (15)
- 9: Compute truncated gamma $\gamma'(l)$ using Eq. (17)
- 10: Apply pixel-wise mapping $T(l)$ to obtain I_e
- 11: **else**
- 12: $I_e = I$ ▷ No enhancement for normal images
- 13: **end if**
- 14: **return** I_e

3.3.3. Comparator: Dual Illumination Estimation for Exposure Correction

As a complementary method to our gamma correction, we are also going to be testing *Dual Illumination Estimation* by Zhang *et al.* [21], which performs simultaneous correction of both underexposed and overexposed regions using a pair of illumination maps. The algorithm depicts an image as $I = R \times L$, where R is the reflectance and L the illumination. For dark regions, it estimates L directly with spatially adaptive smoothness priors, while for bright regions, it processes the inverted image $I' = 1 - I$ in addition to mixing both corrected results with multi-exposure image fusion. The final enhanced image is a locally weighted combination of forward and reverse corrected outputs:

$$I_{\text{final}} = \omega_f I'_f + \omega_r I'_r + (1 - \omega_f - \omega_r)I, \quad (18)$$



Fig. 5: An original dark image of a car (top), the resulting image after applying the IAGCWD algorithm using the default parameters (middle), and the result of applying the Fast-ALPR pipeline on the processed image (bottom). When attempting to apply Fast-ALPR on the original dark image, no license plates were detected. The ground truth is 70NX850.



Fig. 6: An original foggy image (top-left) and the result after applying the DCP-based method (top-right). The bottom-left and bottom-right images show the results of Fast-ALPR on the foggy and de-hazed images, respectively.

where ω_f, ω_r are exposure-dependent weights calculated from contrast, saturation, and local brightness measures.

This dual-path design enables balanced enhancement in different lighting conditions. We will use the publicly available implementation by Attaiki [22] to compare with the IAGCWD method. Despite being more computational, Dual Illumination Estimation is good in that it is able to recover both overexposed and underexposed regions at the same time, which makes it an interesting benchmark to compare against Adaptive Gamma Correction.

3.4. Defogging

3.4.1. Classical Approach: Dark Channel Prior

Using the dark channel prior (DCP) from He et al. [3] for single-image de-hazing, we will be able to compute a dark channel over 15×15 patches and pick the atmospheric light A from the brightest 0.1% of pixels. The transmission is estimated at $t(x) = 1 - 0.95 \cdot DC(I/A)$, and we have refined it with a guided filter [23]. We attain the de-hazed result from:

$$J(x) = \frac{I(x) - A}{\max(t(x), 0.1)} + A. \quad (19)$$

We will subsequently add Contrast Limited Adaptive Histogram Equalization (CLAHE) post-processing (8×8 tiles with clip limit 2.0) in order to increase the quality of the local contrast. DCP typically removes haze well, and runs fast, but is known to create halos around the edges and over-saturate bright regions.

3.4.2. Learning Comparator: DehazeNet

We also test DehazeNet [24] as a baseline, which is a compact CNN that learns a nonlinear mapping between hazy in-

put patches and their respective transmission maps. The advantage of this is that there are fewer halos and more natural color. However, it is slower when it comes to higher resolution images, and needs pretrained weights.

3.5. Evaluation Metrics and Expected Outcome

The effectiveness of each restoration algorithm is evaluated using both image quality metrics and downstream recognition performance. Let I_{in} denote the degraded input image, I_{rest} the restored image, and I_{ref} a pseudo-reference image representing the best visually exposed frame of the same scene when available.

Peak Signal-to-Noise Ratio (PSNR) is used to quantify the mean-square deviation between the restored and reference images,

$$\text{PSNR} = 10 \log_{10} \left(\frac{L^2}{\text{MSE}} \right), \quad (20)$$

$$\text{MSE} = \frac{1}{MN} \sum_{x,y} (I_{rest}(x,y) - I_{ref}(x,y))^2. \quad (21)$$

where $L = 255$ for 8-bit images. Higher PSNR values correspond to smaller reconstruction error and improved luminance fidelity. To assess local geometric and contrast preservation, the Structural Similarity Index (SSIM) is computed over small spatial windows:

$$\text{SSIM}(I_{rest}, I_{ref}) = \frac{(2\mu_r\mu_t + C_1)(2\sigma_{rt} + C_2)}{(\mu_r^2 + \mu_t^2 + C_1)(\sigma_r^2 + \sigma_t^2 + C_2)}, \quad (22)$$

where μ_r, μ_t denote the mean luminances, σ_r^2, σ_t^2 their variances, and σ_{rt} their covariance within the window. Constants C_1, C_2 stabilize the division for small denominators. Because reference images are not always available for real scenes, we additionally employ the Natural Image Quality Evaluator (NIQE) and the Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE), both based on natural scene statistics (NSS). Given an image I_{rest} , NIQE computes a multivariate Gaussian model of normalized luminance coefficients $f(x, y)$ extracted over patches, and evaluates the Mahalanobis distance between its distribution parameters μ_r, Σ_r (from pristine natural images) and those of the test image μ_t, Σ_t :

$$\text{NIQE}(I_{rest}) = \sqrt{(\mu_r - \mu_t)^\top \left(\frac{\Sigma_r + \Sigma_t}{2} \right)^{-1} (\mu_r - \mu_t)}. \quad (23)$$

Lower NIQE values imply better perceived naturalness. BRISQUE operates similarly but uses a support vector regression model to predict perceptual distortion levels from NSS features; smaller values indicate more natural visual quality.

Finally, the restored images are passed through the Fast-ALPR pipeline, and we measure the downstream recognition

accuracy. Let $\hat{T}(I_{\text{rest}})$ denote the OCR output string and T_{gt} the ground-truth plate text. We compute the normalized Levenshtein similarity:

$$A_{\text{OCR}} = 1 - \frac{d_{\text{lev}}(\hat{T}(I_{\text{rest}}), T_{\text{gt}})}{\max(|\hat{T}|, |T_{\text{gt}}|)}, \quad (24)$$

where d_{lev} is the edit distance. For a given OCR detection, $A_{\text{OCR}} = 0$ if the detection perfectly matches the ground truth.

As our expected outcome, we hypothesize that the deblurring and undarkening algorithms will give the largest increase in both PSNR and A_{OCR} , since motion blur and low illumination directly degrade character readability. Defogging is expected to produce modest but consistent improvement in SSIM and BRISQUE by restoring edge contrast and color saturation. Across all degradation conditions, we can expect a correlation between perceptual improvement (low NIQE, high SSIM) and ALPR recognition accuracy.

4. DATASET

4.1. Original Public Dataset

The `fast-alpr` GitHub repository [16] relies on the multi-source dataset collection from LocalizadorPatentes [25], which has multiple public license plate datasets. The combined training data comes from the following sources:

- *OpenImages* [26]: A large-scale dataset containing over 9 million annotated images with 600 object categories. The subset used for ALPR training includes scenes containing vehicles and visible license plates.
- *Romanian License Plate Dataset*: A small, high-quality public dataset with approximately 534 images of European-style Romanian plates.
- *OpenALPR Benchmark Dataset*: The benchmark dataset released by the OpenALPR project, comprising labeled vehicle images with license plate bounding boxes and corresponding text strings.
- *CCPD (Chinese City Parking Dataset)* [27]: A large-scale dataset introduced at ECCV 2018 with approximately 282,000 images, each annotated with the plate bounding box, four corner points, and text.

During the original training of the YOLO-based detection model within `fast-alpr`, all datasets were unified to the YOLO annotation format (x, y, w, h) with input sizes of 448×448 for the full model and 608×608 for the lightweight variant.

4.2. Custom Dataset Collection

We are creating our own dataset of real-world vehicle license plates captured using smartphone cameras and dashcams across multiple U.S. cities (San Diego, San Francisco, Los Angeles, Phoenix, San Jose, Chicago, and Pittsburgh). Each image is categorized into one of four environmental conditions: (1) Normal daylight, (2) Foggy weather, (3) Low-light or nighttime, and (4) Motion blur (vehicle or camera movement). Our goal is to have approximately 300 images per category. Figure 7 shows a few example images for each type of distortion that we plan to analyze.

4.3. Synthetic Augmentation

To expand the dataset and enable controlled experiments, we also plan to generate degraded samples using image-processing models. Motion blur is simulated through a randomized convolutional kernel generator that produces both curved and linear trajectories, optionally introducing per-channel perturbations to mimic real sensor motion. Each kernel is applied via two-dimensional convolution, followed by Gaussian noise injection and alpha blending with the original image to preserve natural appearance. In addition, fog effects will be synthesized using a combination of classical scattering-based models and recent deep learning fog-generation tools capable of controlling density and depth-dependent light scattering. These augmentations allow us to vary the intensity of degradation in a reproducible way while maintaining visual realism, enabling systematic ablation studies on how defogging, undarkening, and deblurring algorithms affect both image quality and downstream license-plate recognition performance.

5. CONCLUSION

We explored how classical image processing techniques can meaningfully improve the visibility and readability of car license plate images captured under difficult conditions, such as fog, low light, and motion blur.

Using our custom dataset collected across several U.S. cities, we analyzed how three restoration algorithms (dark channel prior defogging, adaptive gamma correction, and blind motion deblurring) affect both perceptual image quality and downstream license-plate recognition. Each method addresses a different type of degradation: haze reduces global contrast, low light suppresses details, and motion blur erases texture. Using these restoration algorithms, we hope to show substantial gains in clarity and OCR accuracy.

Our preliminary results suggest that classical algorithms still hold strong practical value, especially when interpretability, reliability, and speed matter more than raw perceptual realism. Moving forward, we plan to extend this study by combining these approaches with lightweight neural compo-



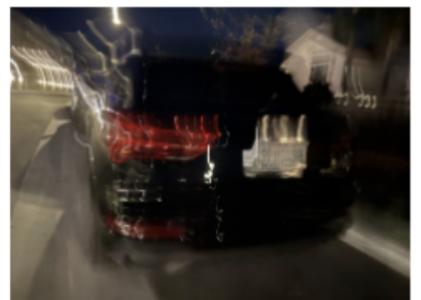
(a) Clear daylight conditions.



(b) Foggy conditions.



(c) Low-light / night conditions.



(d) Motion blur conditions.

Fig. 7: Example samples from our collected dataset showing four real-world imaging conditions: (a) clear daylight, (b) fog, (c) low-light/night, and (d) motion blur. Each row shows three representative examples.

nents, aiming for a balance between physical understanding and data-driven adaptability in future restoration pipelines.

6. REFERENCES

- [1] Vriza Wahyu Saputra, Nanik Suciati, and Chastine Faticahah, “Low light image enhancement in license plate recognition using uretinex-net and trba,” *Procedia Computer Science*, vol. 234, pp. 404–411, 2024, Seventh Information Systems International Conference (ISICO 2023).
- [2] Gang Cao, Lihui Huang, Huawei Tian, Xianglin Huang, Yongbin Wang, and Ruicong Zhi, “Contrast enhancement of brightness-distorted images by improved adaptive gamma correction,” *CoRR*, vol. abs/1709.04427, 2017.
- [3] Kaiming He, Jian Sun, and Xiaou Tang, “Single image haze removal using dark channel prior,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1956–1963.
- [4] Haoyan Gong, Yuzheng Feng, Zhenrong Zhang, Xianxu Hou, Jingxin Liu, Siqi Huang, and Hongbin Liu, “A dataset and model for realistic license plate deblurring,” 2024.
- [5] Kihyun Na, Junseok Oh, Youngkwan Cho, Bumjin Kim, Sungmin Cho, Jinyoung Choi, and Injung Kim, “Mf-lpr: Multi-frame license plate image restoration and recognition using optical flow,” *Computer Vision and Image Understanding*, vol. 256, pp. 104361, May 2025.
- [6] William Hadley Richardson, “Bayesian-based iterative method of image restoration,” *Journal of the Optical Society of America*, vol. 62, no. 1, pp. 55–59, 1972.
- [7] L. B. Lucy, “An iterative technique for the rectification of observed distributions,” *The Astronomical Journal*, vol. 79, no. 6, pp. 745–754, 1974.
- [8] Tony F. Chan and Chiu-Kwong Wong, “Total variation blind deconvolution,” *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998.
- [9] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum, “Image deblurring with blurred/noisy image pairs,” *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, pp. 1–10, 2007.
- [10] Anat Levin, Yair Weiss, Frédo Durand, and William T. Freeman, “Understanding and evaluating blind deconvolution algorithms,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009, pp. 1964–1971.
- [11] Daniele Perrone and Paolo Favaro, “Total variation blind deconvolution: The inverse problem of the inverse problem,” *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4808–4821, 2014.

- [12] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman, “Removing camera shake from a single photograph,” in *ACM SIGGRAPH*. ACM, 2006, vol. 25, pp. 787–794.
- [13] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce, “Non-uniform deblurring for shaken images,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 491–498.
- [14] Valfride Nascimento, Gabriel E. Lima, Rafael O. Ribeiro, William Robson Schwartz, Rayson Laroça, and David Menotti, “Toward advancing license plate super-resolution in real-world scenarios: A dataset and benchmark,” *Journal of the Brazilian Computer Society*, vol. 31, no. 1, pp. 435–449, June 2025.
- [15] Haoyan Gong, Zhenrong Zhang, Yuzheng Feng, Anh Nguyen, and Hongbin Liu, “Lp-diff: Towards improved restoration of real-world degraded license plate,” in *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 17831–17840.
- [16] A. Andrew, “Fast-alpr: End-to-end license plate detection and ocr,” <https://github.com/ankandrew/fast-alpr>, 2024.
- [17] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao, “Yolov9: Learning what you want to learn using programmable gradient information,” *arXiv preprint arXiv:2402.13616*, 2024.
- [18] David S. C. Biggs and Mark Andrews, “Acceleration of iterative image restoration algorithms,” *Applied Optics*, vol. 36, no. 8, pp. 1766–1775, 1997.
- [19] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [20] Vlad Karpushin, “Motion deblur (opencv tutorial implementation),” https://github.com/VladKarpushin/motion_deblur, 2020.
- [21] Qing Zhang, Yongwei Nie, and Wei-Shi Zheng, “Dual illumination estimation for robust exposure correction,” 2019.
- [22] Souhaib Attaiki, “Low light image enhancement,” <https://github.com/pvnieso/Low-light-Image-Enhancement>, 2020.
- [23] Kaiming He, Jian Sun, and Xiaou Tang, “Guided image filtering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.
- [24] Bolun Cai, Xiangmin Xu, Kui Jia, Chunmei Qing, and Dacheng Tao, “Dehazenet: An end-to-end system for single image haze removal,” *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.
- [25] “Localizadordpatentes: Datasets and training scripts for license plate detection,” <https://github.com/ankandrew/LocalizadordPatentes>, 2023.
- [26] Ivan Krasin, Tom Duerig, Neil Alldrin, and et al., “Openimages: A public dataset for large-scale multi-class object detection,” *Google Research*, 2017.
- [27] Zheng Xu, Wei Yang, Li Meng, and et al., “Ccpd: Chinese city parking dataset for license plate detection and recognition,” 2018.