

Article

Text-Prompt-Based AI-Generated Virtual Image Augmentation for Data-Scarce Flood Detection in Urban Flood-Prone Areas

Hanseon Joo and Ook Lee *

Department of Information Systems, Hanyang University, Seoul 04763, Republic of Korea;
jooking012@hanyang.ac.kr

* Correspondence: ooklee@hanyang.ac.kr

Abstract

Urban flood detection requires visual examples of flooded streets and alleys, but such event-state images are difficult to collect at scale. This study examines whether sparse real-image training sets can be strengthened using text-prompt-only AI-generated virtual imagery for ground-level flood detection in flood-prone urban areas. Building on AlleyFloodNet, the generated images were used only as condition-specific training augmentation, while validation and testing were conducted exclusively on real images. Across eight ImageNet-pretrained architectures and three random seeds, the results show that virtual imagery is not a substitute for real flood observations. When virtual images dominated the training set, performance declined. However, when a sufficient real-image anchor was available, virtual augmentation improved the highest observed fixed-test performance. The strongest mixed-condition result was obtained by EfficientNet-B2 under Real30_Aug70, reaching 91.00% accuracy, 89.48% flooded-class recall, and 90.79% macro-F1. These findings suggest that prompt-only virtual imagery can help mitigate real-image scarcity in selected training conditions, but its benefit depends on the real-to-virtual composition and model architecture.

Keywords: flood detection; image classification; AI-generated imagery; synthetic data

1. Introduction

Flood risk in cities is intensifying as climate change, rapid urbanization, impervious-surface expansion, and drainage constraints increase exposure to pluvial and flash-flood events [1,2]. This problem is especially acute in flood-prone urban areas, including low-lying streets, narrow alleys, underpasses, basement or semi-basement entrances, and neighborhoods where residents have limited capacity to avoid or recover from flood impacts. The need for street-level nowcasting and urban flash-flood experiments monitored by local sensors further supports the importance of near-surface observations for rapid situational awareness [3,4].

Remote-sensing and UAV-based approaches remain indispensable for regional flood mapping and understanding post-disaster scenes [5,6]. However, aerial perspectives often cannot resolve the visual cues that determine immediate danger in dense urban settings, such as curb submergence, walkable passage, local water depth, road-surface reflection, drainage-constrained alleys, and entrance-level inundation. Ground-level images from CCTV, traffic cameras, mobile phones, vehicle cameras, and social media can capture these operational cues directly. Prior studies have shown that ground-level imagery can support



Academic Editors: Ming Zhong,
Xiaohong Yang, Shengmei Yang,
Lu Zhuo and Song Ye

Received: 19 May 2026

Revised: 24 June 2026

Accepted: 8 July 2026

Published: 10 July 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

roadway floodwater detection, flood-depth and water-level estimation, flood-severity assessment, traffic-camera-based inundation recognition, low-light waterlogging monitoring, and nighttime flood-extent identification [7–15]. AlleyFloodNet extends this direction by providing a ground-level dataset designed for rapid flood/non-flood classification in economically vulnerable and flood-prone urban areas [16]. Recent studies further show that satellite Earth Observation, SAR, LiDAR/DEM integration, and UAV-SAR fusion can provide information such as broad flood extent and depth context, while ground-level images add street-scale accessibility and safety cues that overhead sensors may miss [17–19].

Despite this need, data from flood-prone urban areas remain difficult to collect at scale. Ordinary non-flood scenes can be sampled repeatedly, but flooded street scenes are rare, hazardous, and short-lived. Public disaster-image and social-media datasets broaden coverage, but event-state images remain heterogeneous in viewpoint, lighting, label quality, and class balance [10,11,20]. For supervised learning, this becomes a class-imbalance problem: a model can appear accurate while failing on rarer flooded classes, precisely the class that matters most for unsafe routing and emergency response [21,22]. AlleyFloodNet raises the issue that diverse real-world data on flood-prone areas in cities is still insufficient [16].

This data bottleneck motivates the use of AI-generated virtual imagery as training-only augmentation. In this context, augmentation is not intended to replace real disaster evidence or to create synthetic evaluation data. Rather, prompt-only generated images can reduce the dependence of model training on rare real flooded-state images and expose the classifier to additional combinations of alley types, road characteristics, regional streetscape, rainfall, lighting, water appearance, viewpoint, lens condition, pedestrians, vehicles, and urban objects. In this study, generated images are added only to condition-specific training sets, while validation and testing remain entirely real. This design asks whether text-prompt-only virtual imagery can strengthen sparse real training data for urban flood-prone areas without treating generated images as observations of actual flood events.

The main contributions of this paper are summarized as follows:

First, in the context of urban flood/non-flood binary classification based on AlleyFloodNet [16], we propose a domain-specific prompt engineering strategy that strengthens sparse real training sets using purely text-prompt-only AI-generated virtual imagery, completely excluding any image references or visual modifications (such as seed images, control images, masks, inpainting, or image-to-image editing). We document how this strategy effectively synthesizes diverse conditions of flood-vulnerable urban areas, including regional context, street characteristics, weather, lighting, non-UAV perspectives, lens occlusion, and pedestrian or vehicle cues.

Second, we evaluate learning conditions sensitive to the ratio of real and virtual images, demonstrating that virtual imagery can be helpful when a sufficient real-image anchor is present, but can degrade performance when the virtual component dominates the training distribution.

Third, we comparatively analyze the behavior of eight pretrained architectures under sparse real-image training and a synthetic-to-real domain gap. The observed advantage of CNN-based backbones in this experiment is interpreted as condition-specific evidence under the present data, optimization, and evaluation protocol, rather than as a general claim that CNNs are inherently superior to Vision Transformers for flood detection.

2. Related Work

2.1. Ground-Level Flood Detection and Urban Flood Perception

Vision-based flood detection research is divided into two main streams, consisting of remote sensing and ground-level detection. Remote-sensing and UAV studies provide spa-

tially extensive flood maps and post-disaster scene information [5,6], while flash-flood monitoring research emphasizes rapid, street-scale impact assessment and local sensing [3,4]. Ground-level flood detection follows the second pathway by analyzing what people, vehicles, and fixed cameras can observe at the road, alley, or entrance scale. This viewpoint is important for urban flood-prone areas because local danger often depends on small visual details such as curb submergence, ponding around drainage inlets, partially blocked lanes, water entering building thresholds, and the passability of narrow passages.

Early ground-level studies treated flood detection as a roadway and crowdsourced image problem. Sazara et al. developed classification and segmentation methods for detecting floodwater on roadways from mobile-phone or optical-camera images [7], and Witherow et al. examined roadway floodwater detection from crowdsourced images [8]. Subsequent studies shifted focus from binary flood appearance toward more operational measures. Alizadeh Kharazi and Behzadan estimated flood depth from street photographs using image processing and deep neural networks [9]; Chaudhary et al. predicted the water level from social-media images through a multi-task ranking approach [10]; and Pereira et al. assessed flood severity from crowdsourced social-media photographs with deep neural networks [11]. These studies show that ground-level imagery can support not only flood/non-flood classification but also depth, water-level, and severity estimation.

A second development is the use of fixed urban cameras and surveillance imagery for continuous monitoring. Zhong et al. used traffic images and YOLOv4 to identify urban flood inundation depth by recognizing submerged pedestrians and vehicles [12]. Zeng et al. proposed a real-time flooding image recognition model using street surveillance images, DeepLabv3+, and a super-resolution generative adversarial network [13]. Zhao et al. focused on low-light urban waterlogging recognition from surveillance videos and constructed a low-light surveillance flood image dataset [14]. Wan et al. further addressed the nighttime urban flood inundation extent using deep learning and surveillance-camera imagery [15]. Together, these works indicate that ground-level visual sensing is moving from isolated flood photographs toward deployable urban monitoring systems that can support localized warning, routing, and emergency response.

AlleyFloodNet is positioned as a dataset-level response to the same ground-level requirement. Rather than relying on broad aerial post-disaster imagery, it targets flood/non-flood classification in economically vulnerable and flood-prone urban environments with ground-level scenes such as alleys, lowlands, residential entrances, and drainage-constrained streets [16]. This focus is directly relevant to urban flood-prone areas because viewpoint mismatch can reduce the practical value of models trained mainly on overhead or distant imagery when the operational task is to judge local safety at street level. The remaining challenge is that even targeted ground-level datasets face limited real flooded-state samples, especially for rare, dangerous, and short-lived flood conditions.

Recent Earth-observation and multi-sensor flood studies further clarify why ground-level datasets should be viewed as complementary to, rather than competitive with, satellite or UAV products. A recent review of Earth Observation services, 3D data integration, and AI-based flood mapping emphasizes that optical, SAR, DEM, and LiDAR data provide regional flood extent, depth, and emergency-mapping context [17]. Fawakherji and Hashemi-Beni integrated UAV optical imagery with Sentinel-1 SAR for flood detection and mapping, showing that multi-resolution sensor fusion can improve flood-map precision in complex landscapes [18]. Misra et al. used deep learning with 10 years of Sentinel-1 SAR observations to map floods at a global scale, illustrating the operational value of cloud-penetrating satellite radar for wide-area monitoring [19]. These approaches provide macro-scale flood extent and hazard context, whereas ground-level datasets such as Al-

leyFloodNet capture curb-level, entrance-level, road-passability, and human-scale visual cues that are difficult to infer reliably from overhead imagery alone.

2.2. AI-Generated and Synthetic Imagery for Disaster and Flood Data Augmentation

Data augmentation is a common response to limited visual training data, and conventional transformations such as cropping, flipping, rotation, scaling, and color perturbation can improve robustness by increasing variation around existing samples [23]. However, these transformations do not create new urban layouts, drainage-constrained alleys, water-depth appearances, vehicle positions, lighting conditions, or entrance-level flood configurations. Generative modeling provides a broader augmentation route because it can synthesize new visual instances rather than only distort existing images. GANs established adversarial image synthesis as a framework for learning image distributions [24], while diffusion probabilistic models and latent diffusion models expanded high-quality image generation through iterative denoising and latent-space generation [25,26].

Recent work on diffusion-model-based augmentation also emphasizes that generative augmentation should be evaluated not only by apparent image realism, but also by downstream generalization, domain fit, label consistency, computational cost, and comparison with conventional augmentation baselines [27]. This point is directly relevant to flood detection, where synthetic scenes may look plausible while still failing to represent the full hydrologic, geographic, and visual diversity of real events.

In disaster imagery, generative augmentation has been used to address limited event data and class imbalance. DisasterGAN synthesized remote-sensing disaster images and explicitly targeted limited training data and class imbalance in disaster assessment [28]. Eltehewy et al. used GAN-generated images to supplement imbalanced natural-disaster data for classification [29]. These studies support the broader premise that disaster-image analysis often lacks sufficient examples of rare event states and that generated images can be used as a sampling strategy for underrepresented disaster classes.

Flood-related augmentation has followed a similar direction. Pouyanfar et al. proposed adversarial data augmentation for unconstrained flood-event detection [30], and Gautam et al. generated realistic river imagery using deep generative adversarial networks for hydrological image analysis [31]. More generally, synthetic data from modern generative models has been studied for recognition tasks and can improve classification or support domain transfer when used to complement real data rather than replace it [32–34]. These studies motivate examining whether generated imagery can improve model learning in flood contexts where real event-state images are scarce.

For urban flood-prone areas, the practical role of AI-generated imagery is therefore augmentation, not evidence substitution. Prompt-only images can be used to expand training exposure to street-scale combinations that are difficult to collect repeatedly during real flood events, such as alley flooding, curb-level ponding, low-light reflections, and water around entrances or vehicles. The present study evaluates this use case by keeping validation and testing to real images only and by comparing real-to-virtual ratios. The central question is not simply whether generated images appear realistic, but whether they improve real-image flood detection when the available real training set is deliberately sparse. A concise summary of the related work and the positioning of the present study is provided in Table 1.

Table 1. Concise summary of related work and study positioning.

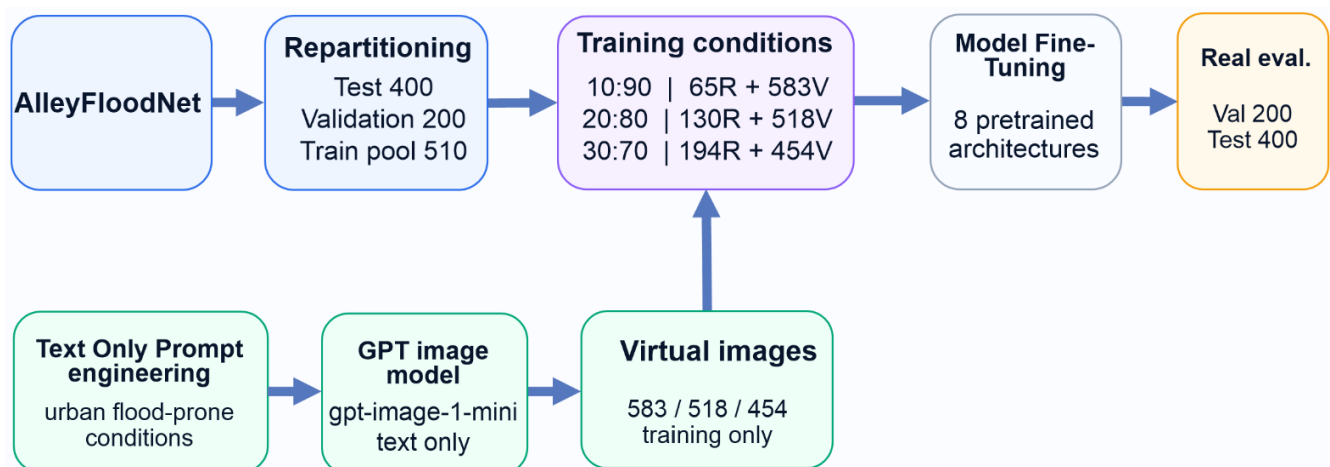
Research Stream	Related Studies	Role in This Study
Ground-level flood detection and datasets	[7–16]	Defines the street- and alley-level flood recognition setting.
Remote sensing/UAV/SAR flood mapping	[5,6,17–19]	Provides complementary macro-scale flood mapping context.
Synthetic augmentation methods	[23–27,32–34]	Motivates generated training data, but requires real-image validation.
Disaster/flood synthetic imagery	[28–31]	Closest prior stream; this study tests prompt-only alley-level augmentation.

3. Materials and Methods

3.1. Study Design

The study was designed as a controlled real-image augmentation experiment. The central experimental variable is the ratio between real training images and text-prompt-only virtual training images. The validation and test sets are composed entirely of real images, and no AI-generated image is used in either evaluation set. This design prevents generated-image artifacts from inflating validation or test performance.

The mixed conditions are denoted using the format RealX_AugY, where X and Y represent the target percentages of real images and AI-generated virtual images, respectively, within a fixed total training capacity of 648 images. For instance, Real10_Aug90 indicates that 10% of the training set consists of real images, while the remaining 90% is filled with virtual images. Specifically, Real10_Aug90 used 583 virtual images, Real20_Aug80 used 518 virtual images, and Real30_Aug70 used 454 virtual images. These quantities correspond to images that actually entered training in each mixed condition. The overall experimental workflow for virtual-image augmentation under sparse real-image conditions is illustrated in Figure 1.

**Figure 1.** Experimental workflow for virtual-image augmentation under sparse real-image conditions.

3.2. Real Dataset Repartitioning

The real-image foundation of the study is AlleyFloodNet [16]. The original data consisted of 1110 images, with 848 images in the original training split and 262 images in the original test split. The class labels are flooded and non-flooded.

For a more conservative evaluation, the test set was enlarged from 262 to 400 real images. The enlarged test set consisted of the original 262 test images plus 138 stratified

real images moved from the original training split. The moved images were removed from all training and validation sets. The final fixed test set contained 209 flooded and 191 non-flooded images. A separate fixed validation set of 200 real images, consisting of 110 flooded and 90 non-flooded images, was then held out for early stopping. After these steps, 510 real images remained available for training, consisting of 282 flooded and 228 non-flooded images. The real dataset partitions used in this study are summarized in Table 2. Figure 2 shows Representative real AlleyFloodNet examples. These images are real dataset samples and were not generated by AI.

Table 2. Real dataset partitions used in the study.

Subset	Total	Flooded	Non-Flooded	Role
Original AlleyFloodNet training split	848	468	380	Original real-image training split before repartitioning
Original AlleyFloodNet test split	262	-	-	Original real-image test split
Fixed real test set	400	209	191	Original test split plus 138 stratified real images moved from training
Fixed real validation set	200	110	90	Held out for early stopping; never used for training
Remaining real training set	510	282	228	Real images available for sparse real-only and mixed training conditions



Figure 2. Representative real AlleyFloodNet examples. These images are real dataset samples and were not generated by AI.

3.3. Prompt Engineering and Text-Prompt-Only Virtual Image Generation

The virtual images used in the reported experiments were generated using the gpt-image-1-mini-image-generation model through text prompts only. No reference image, seed image, conditioning image, control image, segmentation mask, inpainting mask, or image-to-image editing input was provided at any point in the virtual-image generation pipeline. The virtual images were generated to supply the training-only virtual portions of Real10_Aug90, Real20_Aug80, and Real30_Aug70, which used 583, 518, and 454 virtual images, respectively. OpenAI documentation describes GPT Image models as supporting image generation from prompts, with configurable size, quality, output format, and compression options [35].

The prompt template was built as a modular scene description rather than a single fixed sentence. Each prompt combined a region, place type, street characteristic, rainfall or weather condition, time of day, brightness, water-surface state, viewpoint, foreground objects, and class condition. The region component covered East Asia, Europe, the United States, South America, broader North American settings, Southeast Asia, India, and Pakistan. This regional variation was used to avoid a single city style and to produce street scenes with different building facades, road materials, signage, and urban objects.

Place types were selected to match urban flood-prone locations rather than wide regional flood scenes. The template included downtown, residential, and commercial alleys; old low-rise residential areas; narrow back streets; side streets; alleys near local markets; suburban neighborhood roads; roads beside empty lots, riversides, or drainage canals; roads in front of low-rise shops; parking-lot entrance roads; underpass entrances; underground parking-garage entrances and interiors; underground shopping-arcade entrances; service roads behind major boulevards; and narrow or semi-open neighborhood streets. Road width was intentionally constrained because the target task concerns alley-level and local-street-level flood perception.

Street characteristic prompts introduced details that frequently appear in real ground-level photographs. These details included shop signs, neon signage, closed shutters, small eateries, scooters, parked vehicles, market stalls, walls, fences, stairs, slopes, retaining walls, balconies, low-rise shops, scattered buildings, and everyday urban clutter. Weather and illumination prompt varied heavy rain, light rain, wet conditions, and bright non-rain or post-rain scenes. Time of day and brightness were also varied because flood images are affected by shadow, glare, exposure, headlights, and artificial lighting.

Water appearance and viewpoint were controlled explicitly. Prompts separated clear water surfaces from muddy or turbid water and varied reflections on wet pavement. The viewpoint was kept non-UAV and ground-oriented, including human-eye height, vehicle-camera height, fixed surveillance-camera height, and elevated but non-aerial building viewpoints. Some prompts included lens obstructions, such as raindrops, humidity haze, or minor water streaks, to approximate real camera conditions during rainfall. Pedestrians were included in some prompts, such as scale cues; when they were used, water reaching the ankle or higher was categorized as flooded, while shallower water below the ankle was categorized as non-flooded.

To reduce unrealistic outputs, a fixed negative prompt excluded fantasy, surreal, science-fiction, apocalyptic collapse, giant tsunami waves, impossible water physics, distorted buildings or vehicles, duplicated objects, cartoon, illustration, anime, painting styles, excessive blur, low detail, oversaturated cinematic effects, movie-poster composition, fisheye or extreme wide-angle distortion, warped perspective, impossible reflections, malformed text or signage, oversharpened edges, overexposed highlights, underexposed shadows, low-resolution artifacts, compression artifacts, and broken anatomy. The purpose was to keep the generated images close to plausible ground-level photographs while still increasing training diversity.

This prompt-engineering design is a methodological strength of the study because it represents diverse conditions of urban flood-prone areas as structured training scenarios rather than as generic flood images. The prompt strategy did not attempt to reproduce a specific real location. It treated text prompts as a controlled way to broaden the training set across plausible urban contexts, flood depths, lighting states, water appearances, objects, and camera conditions. The generated images were used as virtual training scenarios only. Representative examples of the text-prompt-only AI-generated virtual images used for training data augmentation are shown in Figures 3 and 4. Specifically, Figure 3 presents virtual flood image examples, whereas Figure 4 presents virtual non-flood image examples.



Figure 3. Representative text-prompt-only AI-generated virtual flood image examples used for training data augmentation.

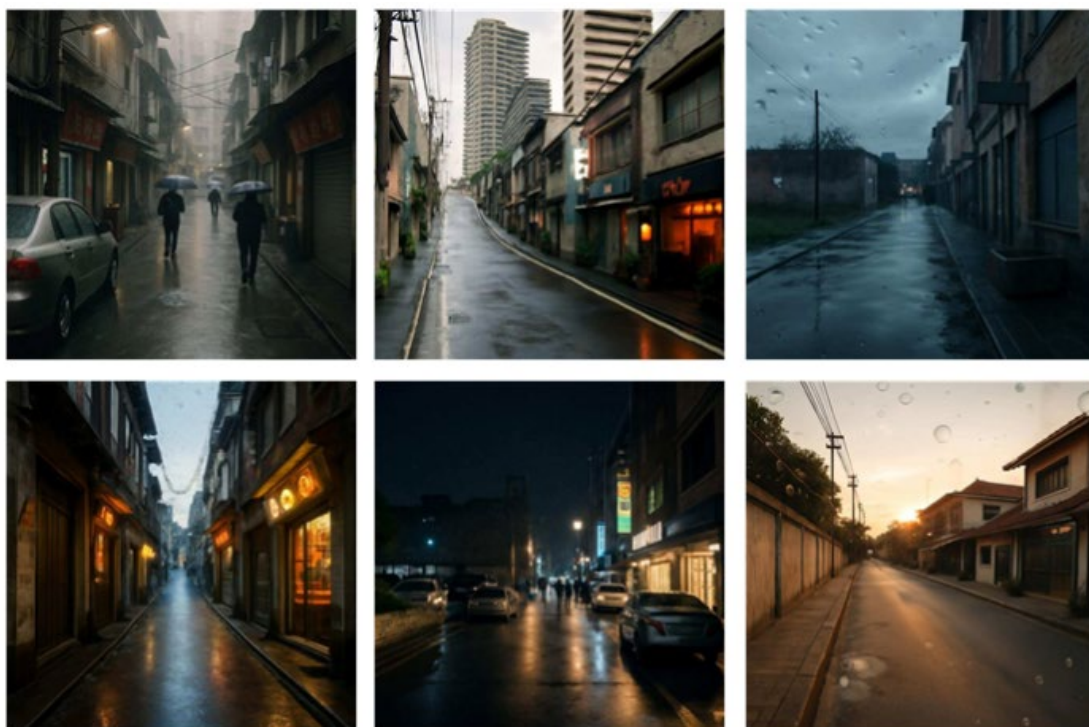


Figure 4. Representative text-prompt-only AI-generated virtual non-flood image examples used for training data augmentation.

The generation and post-processing settings used for the virtual training images are summarized in Table 3. Each generated file used the prefix augalley. Images were requested at 1024×1024 pixels with medium quality and were subsequently post-processed to a maximum side length of 512 pixels. In a sampled generation run, the average image-generation

time was 17.24 s/image, and the average final file size was 55.5 KB/image. These values are reported as practical pipeline descriptors rather than as a full scalability benchmark.

Table 3. Text-prompt-only image generation, prompt engineering, and post-processing settings.

Parameter	Value
Image generation model	gpt-image-1-mini
Requested size	1024 × 1024 pixels
Requested quality	Medium
Output format	JPEG
Output compression	55
File prefix	Augalley_
Post-processing maximum side	768 pixels
Target file size	Approximately 220 KB
JPEG quality bounds after saving	28 to 72
Input modality	Text prompt only
Excluded generation inputs	No reference image, seed image, control image, mask, inpainting input, or image-to-image editing input
Prompt design dimensions	Region; place type; street characteristic; road width and spatial layout; rainfall/weather; time of day and brightness; water appearance and surface condition; viewpoint; lens obstruction and visibility degradation; people, vehicles, signage, and urban objects; flooded or non-flooded state;
Negative prompt purpose	To exclude unrealistic disaster scenes, stylized rendering, distorted geometry, malformed signage, exposure errors, and image artifacts
Average image-generation time (sampled run)	17.24 s/image
Average final file size (sampled run)	55.5 KB/image

3.4. Experimental Training Conditions

The mixed training conditions were constructed by varying the amount of real training imagery and filling the remaining fixed training capacity with prompt-only virtual imagery. The total size of each mixed training set was fixed at 648 images, corresponding to the original 848-image training split after retaining 200 images for validation. Three mixed conditions were evaluated: Real10_Aug90, Real20_Aug80, and Real30_Aug70. These conditions used approximately 10%, 20%, and 30% real training imagery, respectively, with the remaining training slots filled by text-prompt-only virtual images.

For each fixed 648-image mixed training set, the number of virtual images was determined as the residual after selecting the specified number of real images. Real10_Aug90 used 65 real images and 583 text-prompt-only virtual images; Real20_Aug80 used 130 real images and 518 text-prompt-only virtual images; and Real30_Aug70 used 194 real images and 454 text-prompt-only virtual images. These are the only AI-generated images reported as part of the training data for the corresponding mixed conditions.

Training membership was determined by a reproducible file-ordering rule rather than by ad hoc manual choice. For each condition, class counts were first allocated in proportion to the available source distribution. Within each class, real and virtual-image file lists were sorted by filename, and the required number of images was selected deterministically under each predefined random seed setting. The resulting training manifest records the

experiment name, source type, index, file path, filename, numeric label, and class name for every training image used in each condition. The experimental training conditions and condition-specific virtual-image usage are summarized in Table 4.

Table 4. Experimental training conditions and condition-specific virtual-image usage.

Condition	Training Total	Flooded	Non-Flooded	Virtual Used	Description
Real10_only	65	36	29	0	Real-only; no virtual
Real10_Aug90	648	328	320	583	65 real + 583 virtual
Real20_only	130	72	58	0	Real-only; no virtual
Real20_Aug80	648	331	317	518	130 real + 518 virtual
Real30_only	194	107	87	0	Real-only; no virtual
Real30_Aug70	648	334	314	454	194 real + 454 virtual
Real100_only	510	282	228	0	510 real; no virtual

3.5. Model Architectures and Training Procedure

Eight ImageNet-pretrained models were evaluated. ImageNet pretraining was used because large-scale visual pretraining can provide transferable feature representations for downstream visual recognition tasks [36,37]. The tested models were AlexNet, VGG16-BN, ResNet50, DenseNet121, EfficientNet-B2, ConvNeXt-Large, ViT-B/16, and Swin-T [38–45]. The final classifier of each model was replaced with a single-logit binary head for flood/non-flood classification.

All real and virtual images were processed through the same deterministic image pipeline before being supplied to the classifiers: resize to 224×224 pixels, conversion to a tensor, and normalization with ImageNet mean values of 0.485, 0.456, and 0.406 and standard deviations of 0.229, 0.224, and 0.225. No additional conventional image augmentation, such as random cropping, flipping, rotation, color jitter, MixUp, or CutMix, was applied. This choice kept the experimental focus on dataset-level augmentation through condition-specific text-prompt-only virtual images rather than on standard photometric or geometric transformations. Models were trained with BCEWithLogitsLoss and the Adam optimizer. The learning rate was 0.001, the batch size was 64, the maximum number of epochs was 30, and early stopping was applied with a patience value of 4 based on validation loss. For each architecture and training condition, the checkpoint with the lowest validation loss was retained as the best checkpoint; the fixed test set was not used for checkpoint selection, early stopping, or hyperparameter tuning, and was used only once for final performance reporting within each run. Each architecture-condition experiment was repeated using random seeds 36, 42, and 48, and the reported performance values are presented as mean $\pm$ standard deviation over the three runs.

3.6. Evaluation Metrics

Model performance was assessed using accuracy, flooded-class precision, flooded-class recall, and macro-F1. Accuracy provides a broad indication of how often the model assigns the correct label across the test set. However, overall correctness alone does not fully describe model behavior, especially when the two classes are not equally represented. Therefore, precision and recall were reported specifically for the flooded class, which is the focal class of this study. Flooded-class precision reflects the reliability of flooded predictions by measuring how many images predicted as flooded were truly flooded. Flooded-class recall reflects the model's ability to identify actual flooded images. Macro-F1 was used as the overall F1-based performance indicator because it averages class-wise F1 values

and therefore provides a more balanced summary of binary classification behavior than a single-class-specific F1 score.

4. Results

4.1. Model-Wise Fixed-Test Performance Across Training Conditions

The fixed-test results were reorganized at the model level rather than averaged across architectures. Because the eight backbones differed substantially in sparse-data behavior, the analysis first reports the complete result pattern for each model across the all-real reference condition, matched sparse real-only baselines, and text-prompt-only virtual-image augmentation conditions. In Tables 5–12, flooded-class precision, flooded-class recall, and macro-F1 are reported as the mean $\pm$ standard deviation across random seeds 36, 42, and 48. AlexNet reached 0.6933 accuracy and 0.6930 macro-F1 in Real100_only_510. In the sparse settings, it recorded 0.5592/0.5010 in Real10_only and 0.5283/0.4947 in Real10_Aug90, 0.5825/0.5552 in Real20_only and 0.5700/0.5185 in Real20_Aug80, and 0.5450/0.5689 in Real30_only and 0.5792/0.6007 in Real30_Aug70. The augmented rows did not uniformly enhance AlexNet performance; the clearest positive change appeared at the 30% real-image setting.

Table 5. AlexNet fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.6933 $\pm$ 0.0427	0.7321 $\pm$ 0.0673	0.6517 $\pm$ 0.1618	0.6930 $\pm$ 0.0499
Real10_only	0.5592 $\pm$ 0.0240	0.6646 $\pm$ 0.2203	0.6546 $\pm$ 0.4375	0.5010 $\pm$ 0.1371
Real10_Aug90	0.5283 $\pm$ 0.0555	0.3894 $\pm$ 0.3404	0.3917 $\pm$ 0.5180	0.4947 $\pm$ 0.1869
Real20_only	0.5825 $\pm$ 0.0631	0.6222 $\pm$ 0.0205	0.4826 $\pm$ 0.2895	0.5552 $\pm$ 0.1000
Real20_Aug80	0.5700 $\pm$ 0.0739	0.7889 $\pm$ 0.1980	0.3427 $\pm$ 0.3758	0.5185 $\pm$ 0.1475
Real30_only	0.5450 $\pm$ 0.0585	0.7146 $\pm$ 0.2159	0.4992 $\pm$ 0.4350	0.5689 $\pm$ 0.0682
Real30_Aug70	0.5792 $\pm$ 0.0851	0.6736 $\pm$ 0.1795	0.6226 $\pm$ 0.4101	0.6007 $\pm$ 0.1081

ConvNeXt-Large remained a strong model in the all-real reference condition, with 0.9258 accuracy and 0.9251 macro-F1 in Real100_only_510. It achieved 0.7483/0.7075 in Real10_only and 0.7383/0.7152 in Real10_Aug90, showing similar but slightly lower accuracy when the 10% real-image subset was dominated by virtual images. At 20% and 30%, it recorded 0.5933/0.5981 in Real20_only, 0.8750/0.8755 in Real20_Aug80, 0.8008/0.7998 in Real30_only, and 0.8900/0.8910 in Real30_Aug70.

Table 6. ConvNeXt-Large fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.9258 $\pm$ 0.0188	0.9407 $\pm$ 0.0288	0.9141 $\pm$ 0.0584	0.9251 $\pm$ 0.0178
Real10_only	0.7483 $\pm$ 0.2031	0.8784 $\pm$ 0.1113	0.6547 $\pm$ 0.5057	0.7075 $\pm$ 0.2684
Real10_Aug90	0.7383 $\pm$ 0.1611	0.7931 $\pm$ 0.0724	0.6427 $\pm$ 0.3653	0.7152 $\pm$ 0.1828
Real20_only	0.5933 $\pm$ 0.2006	0.4411 $\pm$ 0.4241	0.6045 $\pm$ 0.5317	0.5981 $\pm$ 0.2543
Real20_Aug80	0.8750 $\pm$ 0.0050	0.9064 $\pm$ 0.0755	0.8496 $\pm$ 0.0689	0.8755 $\pm$ 0.0039
Real30_only	0.8008 $\pm$ 0.0944	0.8875 $\pm$ 0.1244	0.7356 $\pm$ 0.2666	0.7998 $\pm$ 0.1066
Real30_Aug70	0.8900 $\pm$ 0.0132	0.8705 $\pm$ 0.0446	0.9219 $\pm$ 0.0228	0.8910 $\pm$ 0.0110

DenseNet121 produced consistently strong CNN results. It reached 0.9058 accuracy and 0.9057 macro-F1 in Real100_only_510. The 10% pair moved from 0.7908/0.7877 in

Real10_only to 0.8033/0.7907 in Real10_Aug90. The 20% augmented condition achieved 0.8583/0.8557 compared with 0.8167/0.8187 in Real20_only, and the 30% pair moved from 0.8375/0.8303 in Real30_only to 0.8525/0.8526 in Real30_Aug70.

Table 7. DenseNet121 fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.9058 $\pm$ 0.0250	0.9146 $\pm$ 0.0475	0.9014 $\pm$ 0.0456	0.9057 $\pm$ 0.0250
Real10_only	0.7908 $\pm$ 0.0437	0.8882 $\pm$ 0.1042	0.7016 $\pm$ 0.1865	0.7877 $\pm$ 0.0512
Real10_Aug90	0.8033 $\pm$ 0.0578	0.8544 $\pm$ 0.0472	0.7385 $\pm$ 0.1531	0.7907 $\pm$ 0.0702
Real20_only	0.8167 $\pm$ 0.0429	0.8337 $\pm$ 0.0550	0.8142 $\pm$ 0.1616	0.8187 $\pm$ 0.0486
Real20_Aug80	0.8583 $\pm$ 0.0281	0.8631 $\pm$ 0.0341	0.8589 $\pm$ 0.0064	0.8557 $\pm$ 0.0255
Real30_only	0.8375 $\pm$ 0.0350	0.8766 $\pm$ 0.0477	0.7902 $\pm$ 0.0397	0.8303 $\pm$ 0.0412
Real30_Aug70	0.8525 $\pm$ 0.0164	0.8611 $\pm$ 0.0599	0.8498 $\pm$ 0.0284	0.8526 $\pm$ 0.0162

EfficientNet-B2 reached 0.9300 accuracy and 0.9286 macro-F1 in Real100_only_510. Its 10% real-only baseline was 0.8575/0.8542, whereas Real10_Aug90 declined to 0.8133/0.7868. At 20%, the augmented condition improved over the matched real-only baseline, from 0.8542/0.8388 in Real20_only to 0.8600/0.8623 in Real20_Aug80. At 30%, EfficientNet-B2 showed the highest augmented result, increasing from 0.8625/0.8537 in Real30_only to 0.9100/0.9079 in Real30_Aug70.

Table 8. EfficientNet-B2 fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.9300 $\pm$ 0.0125	0.9410 $\pm$ 0.0131	0.9194 $\pm$ 0.0123	0.9286 $\pm$ 0.0127
Real10_only	0.8575 $\pm$ 0.0087	0.8577 $\pm$ 0.0059	0.8612 $\pm$ 0.0267	0.8542 $\pm$ 0.0119
Real10_Aug90	0.8133 $\pm$ 0.0543	0.8570 $\pm$ 0.0723	0.7716 $\pm$ 0.2024	0.7868 $\pm$ 0.0953
Real20_only	0.8542 $\pm$ 0.0383	0.9316 $\pm$ 0.0129	0.7665 $\pm$ 0.1071	0.8388 $\pm$ 0.0648
Real20_Aug80	0.8600 $\pm$ 0.0347	0.8790 $\pm$ 0.0863	0.8513 $\pm$ 0.0860	0.8623 $\pm$ 0.0333
Real30_only	0.8625 $\pm$ 0.0475	0.8621 $\pm$ 0.0212	0.8630 $\pm$ 0.1172	0.8537 $\pm$ 0.0609
Real30_Aug70	0.9100 $\pm$ 0.0222	0.9256 $\pm$ 0.0074	0.8948 $\pm$ 0.0456	0.9079 $\pm$ 0.0214

ResNet50 achieved 0.8992 accuracy and 0.8975 macro-F1 in Real100_only_510. It improved from 0.7858/0.7862 in Real10_only to 0.8017/0.8020 in Real10_Aug90. The 20% pair also increased slightly, with 0.8125/0.8040 in Real20_only and 0.8200/0.8158 in Real20_Aug80. At 30%, however, ResNet50 decreased from 0.8958/0.8905 in Real30_only to 0.7558/0.7175 in Real30_Aug70.

Table 9. ResNet50 fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.8992 $\pm$ 0.0247	0.9230 $\pm$ 0.0541	0.8755 $\pm$ 0.0094	0.8975 $\pm$ 0.0260
Real10_only	0.7858 $\pm$ 0.0236	0.8225 $\pm$ 0.1174	0.7727 $\pm$ 0.1638	0.7862 $\pm$ 0.0245
Real10_Aug90	0.8017 $\pm$ 0.0388	0.8557 $\pm$ 0.1056	0.7646 $\pm$ 0.1862	0.8020 $\pm$ 0.0454
Real20_only	0.8125 $\pm$ 0.0890	0.9271 $\pm$ 0.0391	0.6939 $\pm$ 0.2011	0.8040 $\pm$ 0.0948
Real20_Aug80	0.8200 $\pm$ 0.0589	0.8466 $\pm$ 0.0399	0.7909 $\pm$ 0.0985	0.8158 $\pm$ 0.0545
Real30_only	0.8958 $\pm$ 0.0113	0.9079 $\pm$ 0.0425	0.8876 $\pm$ 0.0744	0.8905 $\pm$ 0.0189
Real30_Aug70	0.7558 $\pm$ 0.1686	0.9362 $\pm$ 0.0674	0.5828 $\pm$ 0.3778	0.7175 $\pm$ 0.2066

Swin-T reached 0.6050 accuracy and 0.5429 macro-F1 in Real100_only_510, and most sparse-data rows were much weaker. It recorded 0.4792/0.4344 in Real10_only and 0.4775/0.4309 in Real10_Aug90, 0.4958/0.4446 in Real20_only and 0.4917/0.4307 in Real20_Aug80, and 0.4775/0.4309 in Real30_only and 0.5383/0.5404 in Real30_Aug70. These low and unstable mean scores indicate that Swin-T did not provide a reliable classification pattern in this sparse-data setting.

Table 10. Swin-T fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.6050 $\pm$ 0.1129	0.5006 $\pm$ 0.4498	0.3670 $\pm$ 0.3249	0.5429 $\pm$ 0.1924
Real10_only	0.4792 $\pm$ 0.0029	0.4925 $\pm$ 0.5002	0.3365 $\pm$ 0.5746	0.4344 $\pm$ 0.1836
Real10_Aug90	0.4775 $\pm$ 0.0000	0.1592 $\pm$ 0.2757	0.3333 $\pm$ 0.5774	0.4309 $\pm$ 0.1866
Real20_only	0.4958 $\pm$ 0.0236	0.6667 $\pm$ 0.2896	0.6730 $\pm$ 0.5663	0.4446 $\pm$ 0.1748
Real20_Aug80	0.4917 $\pm$ 0.0245	0.1662 $\pm$ 0.2879	0.3054 $\pm$ 0.5290	0.4307 $\pm$ 0.1862
Real30_only	0.4775 $\pm$ 0.0000	0.1592 $\pm$ 0.2757	0.3333 $\pm$ 0.5774	0.4309 $\pm$ 0.1866
Real30_Aug70	0.5383 $\pm$ 0.1054	0.4201 $\pm$ 0.3946	0.4944 $\pm$ 0.5001	0.5404 $\pm$ 0.1881

VGG16-BN achieved 0.8392 accuracy and 0.8390 macro-F1 in Real100_only_510. The 10% augmented condition showed lower accuracy but higher macro-F1, moving from 0.7317/0.7135 in Real10_only to 0.7183/0.7412 in Real10_Aug90. At 20%, macro-F1 increased from 0.5872 to 0.7147, with accuracy increasing from 0.6117 to 0.6925. At 30%, Real30_Aug70 improved to 0.7717/0.7787 compared with 0.5983/0.3937 in Real30_only.

Table 11. VGG16-BN fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.8392 $\pm$ 0.0315	0.8097 $\pm$ 0.0422	0.8996 $\pm$ 0.0599	0.8390 $\pm$ 0.0356
Real10_only	0.7317 $\pm$ 0.0881	0.8732 $\pm$ 0.0860	0.5784 $\pm$ 0.2399	0.7135 $\pm$ 0.1064
Real10_Aug90	0.7183 $\pm$ 0.0465	0.7042 $\pm$ 0.1022	0.8272 $\pm$ 0.1455	0.7412 $\pm$ 0.0057
Real20_only	0.6117 $\pm$ 0.1513	0.6238 $\pm$ 0.1397	0.6140 $\pm$ 0.1511	0.5872 $\pm$ 0.1579
Real20_Aug80	0.6925 $\pm$ 0.0901	0.6954 $\pm$ 0.1616	0.7995 $\pm$ 0.1016	0.7147 $\pm$ 0.0689
Real30_only	0.5983 $\pm$ 0.1839	0.4005 $\pm$ 0.4509	0.2512 $\pm$ 0.4125	0.3937 $\pm$ 0.3855
Real30_Aug70	0.7717 $\pm$ 0.0101	0.8077 $\pm$ 0.1398	0.7751 $\pm$ 0.1674	0.7787 $\pm$ 0.0173

ViT-B/16 showed weak overall accuracy in most settings despite occasional high flooded-class recall. It recorded 0.5525 accuracy and 0.5448 macro-F1 in Real100_only_510, 0.4908/0.4369 in Real10_only and 0.4975/0.5023 in Real10_Aug90, 0.4825/0.4311 in Real20_only and 0.5017/0.4905 in Real20_Aug80, and 0.4850/0.4380 in Real30_only and 0.5125/0.4383 in Real30_Aug70. This pattern supports treating the transformer baseline separately from the strongest CNN-based results in this sparse-data setting.

Table 12. ViT-B/16 fixed-test performance across all training conditions (mean $\pm$ SD across three random seeds).

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.5525 $\pm$ 0.1050	0.3847 $\pm$ 0.3440	0.5642 $\pm$ 0.4961	0.5448 $\pm$ 0.1923
Real10_only	0.4908 $\pm$ 0.0231	0.3325 $\pm$ 0.2888	0.6635 $\pm$ 0.5746	0.4369 $\pm$ 0.1817
Real10_Aug90	0.4975 $\pm$ 0.0346	0.3473 $\pm$ 0.3039	0.5008 $\pm$ 0.5000	0.5023 $\pm$ 0.1644
Real20_only	0.4825 $\pm$ 0.0087	0.1615 $\pm$ 0.2797	0.3246 $\pm$ 0.5622	0.4311 $\pm$ 0.1869
Real20_Aug80	0.5017 $\pm$ 0.0267	0.4255 $\pm$ 0.1535	0.5434 $\pm$ 0.4802	0.4905 $\pm$ 0.1592
Real30_only	0.4850 $\pm$ 0.0066	0.8282 $\pm$ 0.2975	0.3364 $\pm$ 0.5702	0.4380 $\pm$ 0.1852
Real30_Aug70	0.5125 $\pm$ 0.0482	0.4277 $\pm$ 0.3859	0.2674 $\pm$ 0.4264	0.4383 $\pm$ 0.1642

After validation-based checkpoint selection, we descriptively summarized the highest fixed-test mean macro-F1 result observed within each training condition across the evaluated architectures. This comparison is a final test-performance summary of already trained models, not a procedure for checkpoint selection, early stopping, or hyperparameter tuning. The best weights were selected using the validation set, and the fixed test set was used only for final performance reporting. The accompanying flooded-class precision, flooded-class recall, and macro-F1 values describe the final test behavior of the corresponding highest condition-wise observed result, including whether its predictions were more recall- or precision-oriented.

Table 13 reports the highest fixed-test mean macro-F1 result observed in each training condition after validation-based checkpoint selection. EfficientNet-B2 led the all-real reference condition, reaching 0.9300 accuracy and 0.9286 macro-F1 in Real100_only_510. At 10%, EfficientNet-B2 led the real-only condition, whereas ResNet50 led the augmented condition. However, the highest-performing 10% augmented result was lower than the corresponding 10% real-only result: accuracy decreased from 0.8575 in Real10_only to 0.8017 in Real10_Aug90, and macro-F1 decreased from 0.8542 to 0.8020. This suggests that the 90% virtual-image composition was too aggressive under this sparse real-data condition.

Table 13. Descriptive summary of the highest fixed-test mean macro-F1 result in each training condition after validation-based checkpoint selection.

Condition	Accuracy (Mean $\pm$ SD)	Flooded Precision (Mean $\pm$ SD)	Flooded Recall (Mean $\pm$ SD)	Macro-F1 (Mean $\pm$ SD)
Real100_only_510	0.9300 $\pm$ 0.0125	0.9410 $\pm$ 0.0131	0.9194 $\pm$ 0.0123	0.9286 $\pm$ 0.0127
Real10_only	0.8575 $\pm$ 0.0087	0.8577 $\pm$ 0.0059	0.8612 $\pm$ 0.0267	0.8542 $\pm$ 0.0119
Real10_Aug90	0.8017 $\pm$ 0.0388	0.8557 $\pm$ 0.1056	0.7646 $\pm$ 0.1862	0.8020 $\pm$ 0.0454
Real20_only	0.8542 $\pm$ 0.0383	0.9316 $\pm$ 0.0129	0.7665 $\pm$ 0.1071	0.8388 $\pm$ 0.0648
Real20_Aug80	0.8750 $\pm$ 0.0050	0.9064 $\pm$ 0.0755	0.8496 $\pm$ 0.0689	0.8755 $\pm$ 0.0039
Real30_only	0.8958 $\pm$ 0.0113	0.9079 $\pm$ 0.0425	0.8876 $\pm$ 0.0744	0.8905 $\pm$ 0.0189
Real30_Aug70	0.9100 $\pm$ 0.0222	0.9256 $\pm$ 0.0074	0.8948 $\pm$ 0.0456	0.9079 $\pm$ 0.0214

The 20% and 30% settings showed a different pattern. At 20%, EfficientNet-B2 achieved the highest real-only result, with 0.8542 accuracy and 0.8388 macro-F1, whereas ConvNeXt-Large achieved the highest augmented result, with 0.8750 accuracy and 0.8755 macro-F1. At 30%, ResNet50 achieved the highest real-only result, with 0.8958 accuracy and 0.8905 macro-F1, while EfficientNet-B2 achieved the highest performance under Real30_Aug70, reaching 0.9100 accuracy and 0.9079 macro-F1. These results indicate that text-prompt-only virtual imagery did not replace real observations but was associated with improved observed performance when the training set retained a larger real-image anchor.

The complete model-wise tables also show that architecture-averaged scores can obscure architecture-specific behavior in this experiment. Several sparse-data rows, especially for Swin-T, ViT-B/16, and some older CNN settings, achieved high flooded-class recall with low accuracy, suggesting flooded-class overprediction rather than robust classification. The main result should therefore be interpreted conditionally: the 90% virtual-image condition reduced observed performance when virtual imagery dominated the training set, whereas the 20% and 30% real-image settings showed the highest observed performance among the evaluated models. The strongest augmented outcome was observed for EfficientNet-B2 under Real30_Aug70.

4.2. Architecture-Level Comparison Across Vision Models

The condition-wise winners based on three-seed mean macro-F1 show that the effect of virtual imagery was architecture-dependent. EfficientNet-B2 led the all-real reference

condition, Real10_only, Real20_only, and Real30_Aug70. ResNet50 led Real10_Aug90 and Real30_only, and ConvNeXt-Large led Real20_Aug80. This suggests that modern CNN-based representations were comparatively robust across the strongest condition-wise results, but that the leading architecture changed with the real-to-virtual training composition.

The augmented conditions shifted the leading architecture as the amount of real imagery increased. ResNet50 led Real10_Aug90, ConvNeXt-Large led Real20_Aug80, and EfficientNet-B2 led Real30_Aug70 and delivered the strongest augmented performance in the study. This shift is important because it shows that the benefit of prompt-only virtual imagery should not be described as a uniform dataset-level improvement. Instead, the improvement emerged when the real-image anchor was large enough and when the model architecture could exploit the added visual diversity without overfitting for generator-specific artifacts.

AlexNet and VGG16-BN served as older CNN baselines. They did not define the best result in any training condition, but their inclusion is still useful because it shows that the virtual-image protocol should be evaluated across both legacy and modern CNN backbones. ViT-B/16 and Swin-T were also retained in the experiment, but their sparse-data behavior should be interpreted separately rather than averaged into the main conclusion. Their weaker performance in several sparse settings does not imply that transformer-based vision models are generally unsuitable for flood detection; a more limited interpretation is that this experiment combines small real-image subsets, prompt-generated virtual images, and a real-image-only test set, a setting in which transformer-style models may require larger datasets, stronger regularization, or different fine-tuning schedules.

For this reason, the main analysis emphasizes condition-wise best models and architecture-level patterns rather than a single averaged condition score. This presentation provides clearer results: virtual imagery did not rescue the 10% real condition, but it improved the strongest attainable classifier at 20% and especially 30% real-image availability. The complete per-model metric distribution is best placed in the Supplementary Material so that readers can inspect the full model-level spread without allowing weak baselines to dilute the main empirical finding.

5. Discussion

The results support a more precise interpretation of prompt-only virtual-image augmentation. The central claim is not that AI-generated flood imagery generally improves every model or every real-data regime. Rather, the observed benefit was limited to some sparse real-data settings and depended on both the real-to-virtual ratio and the architecture used for fine-tuning. When the real subset was only 10% of the fixed mixed training capacity, the virtual images dominated the training distribution and reduced the best attainable accuracy and macro-F1. When the real subset increased to 20% or 30%, the virtual images acted more plausibly as diversity-enhancing training examples rather than replacements for the real distribution.

This interpretation is stronger than a conclusion based on averaged architecture scores. Averaging AlexNet, VGG16-BN, ResNet50, DenseNet121, EfficientNet-B2, ConvNeXt-Large, ViT-B/16, and Swin-T into a single condition score treats all architectures as equally informative for deployment. In practice, however, the research question is whether a usable flood/non-flood classifier can be trained under sparse real-image conditions, and the best-performing architecture in each condition is therefore a more direct indicator of practical feasibility. The complete model set remains important, but it is better used to interpret robustness and architecture sensitivity than to define the headline result.

The architecture-level pattern also provides a useful boundary for the study. In this particular low-data setting, the strongest and most stable outcomes were obtained

by CNN-based models, especially ConvNeXt-Large, EfficientNet-B2, DenseNet121, and ResNet50. This should be interpreted as condition-specific evidence under the present data volume, regularization choices, training schedule, and real-image-only test protocols, not as a general claim that CNNs are architecturally superior to vision transformers. The transformer baselines are therefore interpreted separately from the headline augmentation result, and their weaker sparse-data performance is better framed as motivation for transformer-specific tuning, stronger augmentation, larger real datasets, or alternative fine-tuning schedules.

The practical implication is that prompt-engineered virtual imagery can reduce real-data scarcity only under controlled conditions. Generated images should remain training-only data, and validation and testing should remain fully real. This separation is essential because it prevents generator-specific artifacts from entering the evaluation pipeline. It also makes the result more defensible: the reported gains in the favorable augmented settings reflect transfer to real test images rather than performance on synthetic or partially synthetic evaluation data. An additional ethical consideration is that synthetic disaster images may be mistaken for real event evidence or may reproduce prompt-induced biases; therefore, this study clearly treated generated images as training-only augmentation, excluded them from validation and testing, and disclosed the use of image-generation tools.

The present ground-level approach should also be interpreted within a hybrid flood-monitoring framework. Satellite EO, SAR, UAV, LiDAR, and DEM-based systems are better suited for macro-scale inundation extent, regional hazard mapping, water-depth estimation, and emergency mapping across large areas [17–19]. Ground-level image classification can add a different layer of information: localized street passability, curb or entrance-level inundation, drainage-constrained alley conditions, pedestrian and vehicle exposure, and human-scale visual evidence that overhead sensors may miss. A practical future system could therefore combine satellite or UAV flood extent maps with ground-level classifiers to link regional flood boundaries with local accessibility and safety conditions. The present study did not construct a GIS- or coordinate-based spatial visualization; future work could connect geocoded ground-level imagery with GIS layers, Google Maps-based visualization, and real-time satellite or SAR products when location metadata are available and can be shared safely.

Several limitations remain. First, the task is binary flood/non-flood classification and does not estimate flood depth, segment inundated pixels, localize water boundaries, or model hydrologic dynamics. Second, the virtual images were generated from text prompts only; this improves procedural reproducibility but may also introduce prompt-induced visual regularities. Third, the results are based on a fixed real-image test set and should be validated on additional urban flood datasets, cities, seasons, camera types, and weather conditions. Fourth, the transformer baselines may require a different optimization strategy before their performance can be compared conclusively with CNN backbones in this sparse-data setting.

Another limitation is that the generated images were not independently evaluated for visual realism or semantic alignment using Fréchet Inception Distance (FID), CLIP-based similarity, expert visual assessment, or a formal human-rating protocol. The study therefore does not claim that the generated images fully reproduce real flood-scene complexity. Because these assessments require a separate quality-evaluation protocol beyond the present controlled classification experiment, they are treated as future work rather than as part of the current analysis. Future work should combine downstream real-image test performance with explicit quality-control measures for generated imagery, including realism, diversity, label consistency, potential bias, and whether prompt-induced artifacts affect model learning [27,32,33]. Future work should also examine prompt-component

ablations to determine whether individual prompt categories, such as weather, lighting, viewpoint, region, or water appearance, contribute differently to image usefulness and downstream classification.

6. Conclusions

This study examined whether text-prompt-only AI-generated virtual imagery can support urban flood/non-flood image classification when real flooded-scene examples are scarce. The model-level results show that virtual imagery is not a substitute for real data. The all-real reference condition remained the strongest overall setting, with EfficientNet-B2 reaching 0.9300 accuracy and 0.9286 macro-F1. The augmented settings should therefore be interpreted conditionally: prompt-only virtual imagery helped in some limited-real-data settings, but only when enough real imagery remained available to anchor the training distribution.

The clearest positive augmented result was Real30_Aug70, where EfficientNet-B2 achieved 0.9100 accuracy, 0.8948 flooded-class recall, and 0.9079 macro-F1. This was the strongest augmented condition, whereas the strongest overall result remained Real100_only_510. Real20_Aug80 also produced a stronger best-model result than its real-only counterpart. By contrast, Real10_Aug90 reduced the best attainable accuracy and macro-F1 relative to Real10_only, showing that excessive virtual-image dominance can be harmful. The main conclusion is therefore conditional: prompt-only virtual imagery can help under data scarcity, but its benefit depends on architecture and real/synthetic composition.

Within the present experimental setting, CNN-based models provided the most stable empirical support for this conclusion. EfficientNet-B2, ResNet50, and ConvNeXt-Large accounted for the condition-wise highest observed fixed-test results, whereas AlexNet, VGG16-BN, DenseNet121, ViT-B/16, and Swin-T did not lead in any condition. This finding should not be generalized as the architectural superiority of CNNs over transformer-based models. Future work should expand real test coverage, evaluate additional prompt-generation strategies, conduct prompt-component ablations, assess generated-image realism using dedicated quality metrics or expert review, provide full per-model supplementary metrics, test transformer-specific fine-tuning protocols, and examine hybrid monitoring workflows that integrate ground-level images with satellite, SAR, UAV, LiDAR, or DEM-derived flood information.

Supplementary Materials: The following supporting information can be downloaded at <https://www.mdpi.com/article/10.3390/urbansci10070398/s1>, Table S1: Full per-model fixed-test metrics reported as mean $\pm$ standard deviation across three experimental runs; File S1: Prompt families and image-generation settings for the text-prompt-only virtual images; File S2: Training code and configuration. The condition-specific real and virtual image training manifest originally described as Table S2 is not provided as a separate supplementary table because the corresponding dataset and manifest information are available through the uploaded Kaggle dataset.

Author Contributions: Conceptualization, H.J. and O.L.; methodology, H.J. and O.L.; software, H.J.; validation, H.J.; formal analysis, H.J.; investigation, H.J. and O.L.; resources, H.J. and O.L.; data curation, H.J.; writing—original draft preparation, H.J.; writing—review and editing, H.J. and O.L.; visualization, H.J.; supervision, O.L.; project administration, O.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable. This study used image datasets and AI-generated virtual imagery and did not involve human subjects or animal experiments.

Informed Consent Statement: Not applicable.

Data Availability Statement: The AlleyFloodNet dataset is described in the cited publication. The condition-specific text-prompt-only virtual images used in the reported training experiments, training manifests, prompt descriptions, and training code can be made available from the corresponding author upon reasonable request, subject to repository preparation, applicable platform terms, and dataset-use restrictions.

Acknowledgments: During the preparation of this manuscript, the authors used OpenAI ChatGPT 5.5 for language drafting, organization, and editorial refinement. The study also used an OpenAI GPT image-generation API to generate text-prompt-only virtual images for training augmentation, as described in the Methods section. No reference, seed, control, mask, or image-to-image input was used for this generation step. The authors reviewed and edited the outputs and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Definition
AI	Artificial Intelligence
APC	Article Processing Charge
CNN	Convolutional Neural Network
F1	F1-score
GAN	Generative Adversarial Network
GPT	Generative Pretrained Transformer
SAR	Synthetic Aperture Radar
UAV	Unmanned Aerial Vehicle
ViT	Vision Transformer

References

1. IPCC. *Climate Change 2022: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*; Portner, H.-O., Roberts, D.C., Tignor, M., Poloczanska, E.S., Mintenbeck, K., Alegria, A., Craig, M., Langsdorf, S., Loschke, S., Moller, V., et al., Eds.; Cambridge University Press: Cambridge, UK, 2022.
2. United Nations Office for Disaster Risk Reduction. *Sendai Framework for Disaster Risk Reduction 2015–2030*; UNDRR: Geneva, Switzerland, 2015.
3. Costabile, P.; Costanzo, C.; Kalogiros, J.; Bellos, V. Toward street-level nowcasting of flash floods impacts based on HPC hydrodynamic modeling at the watershed scale and high-resolution weather radar data. *Water Resour. Res.* **2023**, *59*, e2023WR034599. [\[CrossRef\]](#)
4. Moy de Vitry, M.; Dicht, S.; Leitão, J.P. floodX: Urban flash flood experiments monitored with conventional and alternative sensors. *Earth Syst. Sci. Data* **2017**, *9*, 657–666. [\[CrossRef\]](#)
5. Bentivoglio, R.; Isufi, E.; Jonkman, S.N.; Taormina, R. Deep learning methods for flood mapping: A review of existing applications and future research directions. *Hydrol. Earth Syst. Sci.* **2022**, *26*, 4345–4378. [\[CrossRef\]](#)
6. Rahnemoonfar, M.; Chowdhury, T.; Sarkar, A.; Varshney, D.; Yari, M.; Murphy, R.R. FloodNet: A high resolution aerial imagery dataset for post flood scene understanding. *IEEE Access* **2021**, *9*, 89644–89654. [\[CrossRef\]](#)
7. Sazara, C.; Cetin, M.; Iftekharuddin, K.M. Detecting floodwater on roadways from image data with handcrafted features and deep transfer learning. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 1852–1857.
8. Witherow, M.A.; Sazara, C.; Winter-Arboleda, I.M.; Elbakary, M.I.; Cetin, M.; Iftekharuddin, K.M. Floodwater detection on roadways from crowdsourced images. *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* **2019**, *7*, 529–540. [\[CrossRef\]](#)
9. Alizadeh Kharazi, B.; Behzadan, A.H. Flood depth mapping in street photos with image processing and deep neural networks. *Comput. Environ. Urban Syst.* **2021**, *88*, 101628. [\[CrossRef\]](#)
10. Chaudhary, P.; D’Aronco, S.; Leitão, J.P.; Schindler, K.; Wegner, J.D. Water level prediction from social media images with a multi-task ranking approach. *ISPRS J. Photogramm. Remote Sens.* **2020**, *167*, 252–262. [\[CrossRef\]](#)
11. Pereira, J.; Monteiro, J.; Silva, J.; Estima, J.; Martins, B. Assessing flood severity from crowdsourced social media photos with deep neural networks. *Multimed. Tools Appl.* **2020**, *79*, 26197–26223. [\[CrossRef\]](#)

12. Zhong, P.; Liu, Y.; Zheng, H.; Zhao, J. Detection of urban flood inundation from traffic images using deep learning methods. *Water Resour. Manag.* **2024**, *38*, 287–301. [\[CrossRef\]](#)
13. Zeng, Y.-F.; Chang, M.-J.; Lin, G.-F. A novel AI-based model for real-time flooding image recognition using super-resolution generative adversarial network. *J. Hydrol.* **2024**, *638*, 131475. [\[CrossRef\]](#)
14. Zhao, J.; Wang, X.; Zhang, C.; Hu, J.; Wan, J.; Cheng, L.; Shi, S.; Zhu, X. Urban waterlogging monitoring and recognition in low-light scenarios using surveillance videos and deep learning. *Water* **2025**, *17*, 707. [\[CrossRef\]](#)
15. Wan, J.; Wang, X.; Cheng, Y.; Zhang, C.; Xue, F.; Yang, T.; Tong, F.; Wang, Q.J. Identification of nighttime urban flood inundation extent using deep learning. *Nat. Hazards Earth Syst. Sci.* **2025**, *25*, 4361–4373. [\[CrossRef\]](#)
16. Lee, O.; Joo, H. AlleyFloodNet: A ground-level image dataset for rapid flood detection in economically and flood-vulnerable areas. *Electronics* **2025**, *14*, 2082. [\[CrossRef\]](#)
17. Destefanis, T.; Guliyeva, S.; Boccardo, P.; Fissore, V. Advancing flood detection and mapping: A review of Earth Observation services, 3D data integration, and AI-based techniques. *Remote Sens.* **2025**, *17*, 2943. [\[CrossRef\]](#)
18. Fawakherji, M.; Hashemi-Beni, L. Flood detection and mapping through multi-resolution sensor fusion: Integrating UAV optical imagery and satellite SAR data. *Geomat. Nat. Hazards Risk* **2025**, *16*, 2493225. [\[CrossRef\]](#)
19. Misra, A.; White, K.; Nsutezo, S.F.; Straka, W., III; Lavista, J. Mapping global floods with 10 years of satellite radar data. *Nat. Commun.* **2025**, *16*, 5762. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Alam, F.; Ofli, F.; Imran, M. CrisisMMD: Multimodal Twitter datasets from natural disasters. In Proceedings of the International AAAI Conference on Web and Social Media, Stanford, CA, USA, 25–28 June 2018; Volume 12, pp. 465–473.
21. Krawczyk, B. Learning from imbalanced data: Open challenges and future directions. *Prog. Artif. Intell.* **2016**, *5*, 221–232. [\[CrossRef\]](#)
22. Johnson, J.M.; Khoshgoftaar, T.M. Survey on deep learning with class imbalance. *J. Big Data* **2019**, *6*, 27. [\[CrossRef\]](#)
23. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [\[CrossRef\]](#)
24. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In *Proceedings of the Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014*; NeurIPS: San Diego, CA, USA, 2014; Volume 27.
25. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. In *Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–12 December 2020*; NeurIPS: San Diego, CA, USA, 2020; Volume 33, pp. 6840–6851.
26. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 10684–10695.
27. Alimisis, P.; Mademlis, I.; Radoglou-Grammatikis, P.; Sarigiannidis, P.; Papadopoulos, G.T. Advances in diffusion models for image data augmentation: A review of methods, models, evaluation metrics and future research directions. *Artif. Intell. Rev.* **2025**, *58*, 112. [\[CrossRef\]](#)
28. Rui, X.; Cao, Y.; Yuan, X.; Kang, Y.; Song, W. DisasterGAN: Generative adversarial networks for remote sensing disaster image generation. *Remote Sens.* **2021**, *13*, 4284. [\[CrossRef\]](#)
29. Eltehewy, R.; Abouelfarag, A.; Saleh, S.N. Efficient classification of imbalanced natural disasters data using generative adversarial networks for data augmentation. *ISPRS Int. J. Geo-Inf.* **2023**, *12*, 245. [\[CrossRef\]](#)
30. Pouyanfar, S.; Tao, Y.; Sadiq, S.; Tian, H.; Tu, Y.; Wang, T.; Chen, S.-C.; Shyu, M.-L. Unconstrained flood event detection using adversarial data augmentation. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 155–159.
31. Gautam, A.; Sit, M.; Demir, I. Realistic river image synthesis using deep generative adversarial networks. *Front. Water* **2022**, *4*, 784441. [\[CrossRef\]](#)
32. He, R.; Sun, S.; Yu, X.; Xue, C.; Zhang, W.; Torr, P.; Bai, S.; Qi, X. Is synthetic data from generative models ready for image recognition? In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
33. Azizi, S.; Kornblith, S.; Saharia, C.; Norouzi, M.; Fleet, D.J. Synthetic data from diffusion models improves ImageNet classification. *arXiv* **2023**, arXiv:2304.08466.
34. Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; pp. 23–30.
35. OpenAI. Image Generation—OpenAI API Documentation. Available online: <https://developers.openai.com/api/docs/guides/image-generation> (accessed on 14 May 2026).
36. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.

37. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In *Proceedings of the Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, Montreal, QC, Canada, 8–13 December 2014*; NeurIPS: San Diego, CA, USA, 2014; Volume 27.
38. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
39. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the International Conference on Learning Representations*, San Diego, CA, USA, 7–9 May 2015.
40. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [[CrossRef](#)]
41. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. [[CrossRef](#)]
42. Tan, M.; Le, Q.V. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
43. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, USA, 18–24 June 2022; pp. 11976–11986.
44. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16×16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations*, Virtual, 3–7 May 2021.
45. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical vision Transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.