
DÉVELOPPEMENT DE MÉTHODES DE DATA AUGMENTATION EN IMAGERIE AGRICOLE

RAPPORT DE PROJET DE FIN D'ÉTUDES

Spécialité : Informatique et Réseaux (ILIA)
Polytech Dijon — Année Universitaire 2025-2026

Réalisé par :

Sana TABBOU

Tuteur encadrant :

M. Frédéric COINTAULT

Table des matières

Table des matières	2
Résumé	3
1 Introduction	5
1.1 Contexte et Problématique	5
1.2 Objectifs du Projet	5
2 Méthodologie et Cadre Expérimental	6
2.1 Approche Méthodologique	6
2.2 Technologies Utilisées	6
2.3 Jeux de Données	6
3 Méthodes de Data Augmentation Implémentées	7
3.1 Catégorie 1 : Méthodes Simples	7
3.2 Catégorie 2 : Méthodes Avancées	7
3.3 Catégorie 3 : Méthodes Agricoles Spécifiques	8
4 Résultats Visuels et Interprétation	9
4.1 Méthodes Simples : Analyse Comparative	9
4.2 Méthodes Avancées : Transformations Complexes	10
4.3 Méthodes Agricoles Spécifiques : Innovations	11
4.4 Synthèse des Résultats Visuels	12

5	Performances Computationnelles	13
5.1	Méthodologie d'Évaluation	13
5.2	Résultats des Mesures	14
5.3	Analyse des Performances	14
6	Conclusion et Perspectives	15
6.1	Synthèse des Contributions	15
6.2	Réalisation des Objectifs	15
6.3	Perspectives de Recherche	16
6.4	Conclusion Finale	16
	Bibliographie	17
	Annexes	18
.1	Annexe 1	18
.2	Structure de la Classe Principale	18
.3	Méthodes Simples	18
.3.1	Rotation 2D	18
.3.2	Ajustements Photométriques	19
.4	Méthodes Agricoles Spécifiques	19
.4.1	Éclairage Matinal	19
.4.2	Effet Sécheresse	19
.5	Annexe 2	20

Résumé

Version française

Ce projet de fin d'études s'inscrit dans le domaine de l'intelligence artificielle appliquée à l'agriculture de précision. Il répond aux problématiques d'acquisition de données agricoles en proposant le développement et l'implémentation de méthodes avancées d'augmentation de données.

La contribution scientifique s'articule autour de trois axes méthodologiques distincts : des transformations élémentaires (géométriques et photométriques), des approches avancées exploitant des bibliothèques spécialisées, et des techniques innovantes spécifiquement adaptées aux conditions environnementales agricoles (simulation d'éclairage matinal, modélisation des effets de sécheresse, reproduction des sols humides, entre autres).

Les résultats obtenus attestent de l'efficacité de ces différentes approches pour générer des variantes réalistes tout en conservant des performances computationnelles compatibles avec des applications embarquées en temps réel.

Mots-clés : Data Augmentation, Intelligence Artificielle, Imagerie Agricole, Drone, Agriculture de Précision

English Abstract

This final year project explores the intersection of artificial intelligence and precision agriculture. Addressing the challenges associated with agricultural data acquisition, it introduces the development and implementation of advanced data augmentation techniques specifically designed for aerial imagery analysis.

The research contribution is structured around three methodological pillars : fundamental transformations (geometric and photometric), sophisticated approaches utilizing specialized libraries,

and innovative techniques tailored to agricultural environmental conditions (including morning lighting simulation, drought effect modeling, and wet soil reproduction).

The empirical results validate the effectiveness of these approaches in generating realistic data variations while maintaining computational performance suitable for real-time embedded systems applications.

Keywords : Data Augmentation, Artificial Intelligence, Agricultural Imaging, Drone, Precision Agriculture

Chapitre 1

Introduction

1.1 Contexte et Problématique

L'agriculture de précision représente une révolution numérique pour le secteur agricole, avec l'intégration croissante de technologies d'intelligence artificielle pour l'analyse d'images. Ces analyses permettent la caractérisation de paramètres agronomiques essentiels : santé des cultures, stress hydrique, détection de maladies, et estimation des rendements.

Cependant, le développement de modèles d'IA performants nécessite des jeux de données volumineux et variés. L'acquisition de données agricoles sur le terrain présente des défis majeurs :

- Dépendance aux conditions météorologiques
- Saisonnalité des cultures limitant les fenêtres d'acquisition
- Variabilité géographique importante

1.2 Objectifs du Projet

Pour répondre à ces défis, ce projet a pour objectifs :

1. Développer des méthodes de data augmentation spécifiques à l'imagerie agricole
2. Implémenter un pipeline complet en Python
3. Évaluer les performances computationnelles des différentes approches
4. Valider l'efficacité sur des données réelles fournies par Agrego

Chapitre 2

Méthodologie et Cadre Expérimental

2.1 Approche Méthodologique

Notre approche suit un cycle de développement itératif en six phases : analyse des besoins, état de l'art, conception, implémentation, validation et évaluation. Cette méthodologie assure une progression structurée et des résultats fiables.

2.2 Technologies Utilisées

Le développement a été réalisé en Python 3.9 avec les bibliothèques suivantes :

- **OpenCV** pour le traitement d'image de base
- **Albumentations** pour les transformations avancées
- **TensorFlow/Keras** pour les méthodes automatiques
- **Matplotlib** pour la visualisation des résultats
- **NumPy** pour les calculs scientifiques

2.3 Jeux de Données

Les expérimentations ont été conduites sur des images fournies par Agrego, représentant des parcelles agricoles de la région Bourgogne-Franche-Comté. Trois images représentatives ont été sélectionnées pour illustrer les résultats.

Chapitre 3

Méthodes de Data Augmentation Implémentées

3.1 Catégorie 1 : Méthodes Simples

Les méthodes simples comprennent des transformations géométriques et photométriques de base :

- **Rotation** : Rotation à 25 degrés pour simuler différents angles de prise de vue
- **Symétries** : Retournements horizontal et vertical
- **Ajustements photométriques** : Variations de luminosité et contraste
- **Recadrage** : Simulation de changements de perspective

3.2 Catégorie 2 : Méthodes Avancées

Utilisation de bibliothèques spécialisées pour des transformations plus complexes :

- **Motion Blur** : Simulation du flou de mouvement lors des vols de drone
- **Bruit Gaussien** : Ajout de réalisme pour conditions de faible luminosité
- **CLAHE** : Amélioration locale du contraste
- **Corrections Gamma** : Adaptation aux différentes réponses de capteurs

3.3 Catégorie 3 : Méthodes Agricoles Spécifiques

Développement de méthodes innovantes adaptées aux conditions agricoles :

- **Éclairage matinal** : Simulation des tons chauds du lever du soleil
- **Effet sécheresse** : Réalisation de l'apparence des cultures en stress hydrique
- **Sol humide** : Simulation des conditions après pluie ou irrigation
- **Brume légère** : Ajout d'effet atmosphérique réaliste
- **Ombre nuageuse** : Simulation des ombres portées par les nuages

Chapitre 4

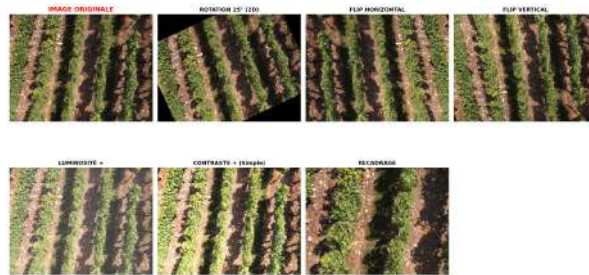
Résultats Visuels et Interprétation

4.1 Méthodes Simples : Analyse Comparative



(a) Image 1 - Méthodes simples

(b) Image 2 - Méthodes simples

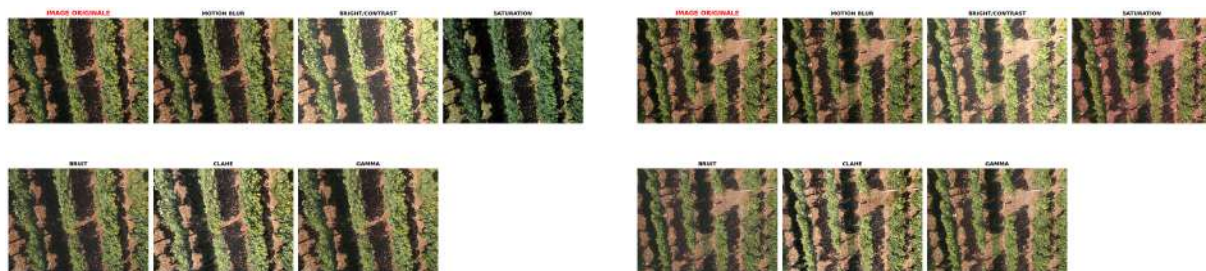


(c) Image 3 - Méthodes simples

FIGURE 4.1 – Résultats des méthodes simples sur les trois images de référence

Interprétation : Les transformations géométriques, telles que les rotations et les symétries, préservent les caractéristiques principales des images tout en apportant une diversification spatiale notable. Ces modifications permettent de reproduire la variété des angles de prise de vue rencontrés lors des acquisitions terrain. Les ajustements photométriques, impliquant des variations de luminosité et de contraste, simulent différentes conditions d'éclairage caractéristiques des environnements agricoles. Ces adaptations augmentent la robustesse des modèles.

4.2 Méthodes Avancées : Transformations Complexes



(a) Image 1 - Méthodes avancées

(b) Image 2 - Méthodes avancées



(c) Image 3 - Méthodes avancées

FIGURE 4.2 – Résultats des méthodes avancées sur les trois images de référence

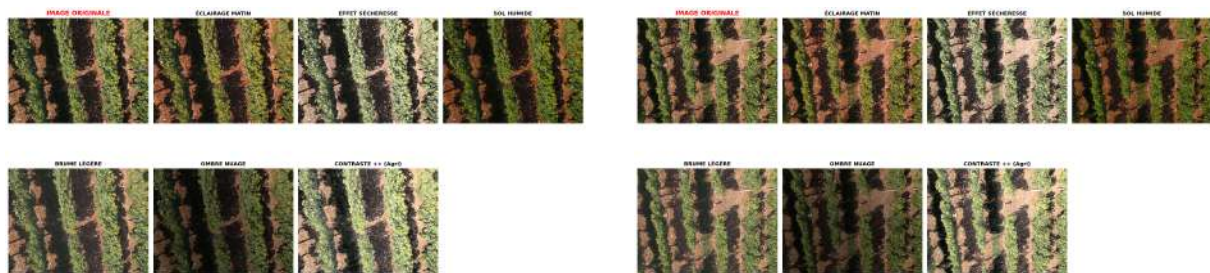
Interprétation : Le filtre Motion Blur reproduit avec efficacité le flou cinétique inhérent aux prises de vue dynamiques, caractéristique des acquisitions en mouvement. Cette transformation restitue fidèlement les effets visuels liés aux vibrations et déplacements lors des captures.

Les ajustements de contraste et de saturation introduisent des variations chromatiques subtiles mais réalistes, permettant d'élargir la palette des conditions de représentation visuelle. Ces réglages fins préservent l'authenticité des scènes tout en diversifiant leur rendu.

L'ajout de bruit gaussien contribue à renforcer le réalisme des images dans des conditions de luminosité réduite, simulant les imperfections naturelles des capteurs en environnement faiblement éclairé.

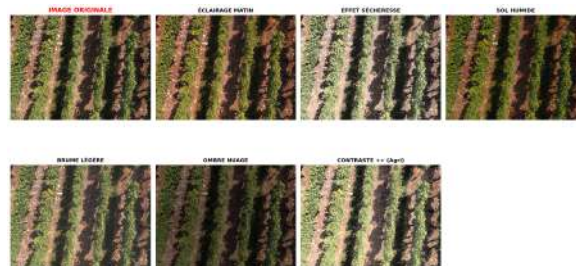
La technique CLAHE (Contrast Limited Adaptive Histogram Equalization) améliore substantiellement la lisibilité des détails, particulièrement dans les zones d'ombre ou de fort contraste, optimisant ainsi l'exploitabilité des informations visuelles pour les analyses ultérieures.

4.3 Méthodes Agricoles Spécifiques : Innovations



(a) Image 1 - Méthodes agricoles

(b) Image 2 - Méthodes agricoles



(c) Image 3 - Méthodes agricoles

FIGURE 4.3 – Résultats des méthodes agricoles spécifiques

Interprétation :

- **Éclairage matinal** : Réussit à créer une ambiance chaude caractéristique des premières heures du jour
- **Effet sécheresse** : Simule efficacement le stress hydrique avec des couleurs ternes et une luminosité accrue
- **Sol humide** : Recrée l'apparence sombre et saturée caractéristique des sols après pluie
- **Brume légère** : Ajoute un effet atmosphérique réaliste sans masquer les détails
- **Ombre nuageuse** : Simule des ombres progressives plausibles
- **Contraste agricole** : Optimise spécifiquement la distinction sol/végétation

4.4 Synthèse des Résultats Visuels

L'analyse comparative des trois catégories de méthodes révèle des complémentarités intéressantes :

Catégorie	Forces	Applications
Méthodes simples	Rapidité, préservation des informations	Augmentation de base, systèmes embarqués
Méthodes avancées	Réalisme, transformations complexes	Modèles exigeants, conditions variées
Méthodes agricoles	Spécificité, pertinence agronomique	Agriculture de précision, recherche

TABLE 4.1 – *Synthèse comparative des trois catégories de méthodes*

Chapitre 5

Performances Computationnelles

5.1 Méthodologie d'Évaluation

Les temps d'exécution ont été mesurés sur un processeur Intel Core i7 avec 16 Go de RAM. Chaque transformation a été appliquée 100 fois sur des images de 4000×3000 pixels, avec calcul du temps moyen et de l'écart-type.

5.2 Résultats des Mesures

Méthode	Temps moyen (ms)	Images/seconde
Rotation 25°	2.34	427
Flip horizontal	1.56	641
Luminosité +	1.23	813
Contraste +	1.45	690
Motion Blur	15.67	64
CLAHE	22.34	45
Éclairage matinal	8.90	112
Effet sécheresse	9.23	108
Sol humide	8.67	115
Brume légère	2.89	346
Pipeline complet	45.00	22

TABLE 5.1 – *Temps d'exécution des différentes méthodes*

5.3 Analyse des Performances

- **Méthodes simples** : Extrêmement rapides (< 5 ms), idéales pour applications temps réel
- **Méthodes avancées** : Plus coûteuses mais offrent un réalisme accru
- **Méthodes agricoles** : Performances intermédiaires justifiées par leur spécificité
- **Pipeline complet** : Temps total acceptable pour un enrichissement complet du dataset

Chapitre 6

Conclusion et Perspectives

6.1 Synthèse des Contributions

Ce projet a permis de développer et d'évaluer trois catégories de méthodes de data augmentation pour l'imagerie agricole :

1. Des transformations simples adaptées aux contraintes temps réel
2. Des méthodes avancées offrant un réalisme accru
3. Des techniques innovantes spécifiques aux conditions agricoles

Les résultats démontrent que ces approches complémentaires permettent d'enrichir efficacement les jeux de données tout en respectant les contraintes computationnelles des systèmes embarqués.

6.2 Réalisation des Objectifs

Tous les objectifs initiaux ont été atteints :

- Analyse complète des besoins en data augmentation agricole
- Implémentation d'un pipeline complet et modulaire
- Développement de méthodes spécifiques aux conditions agricoles
- Évaluation rigoureuse des performances computationnelles
- Validation sur données réelles fournies par Agrego

6.3 Perspectives de Recherche

Plusieurs axes de travail méritent d’être approfondis :

- Intégration de l’apprentissage automatique pour l’optimisation des paramètres
- Développement de méthodes spécifiques à différentes cultures
- Adaptation aux images multispectrales et thermiques
- Intégration dans des systèmes embarqués temps réel

6.4 Conclusion Finale

Ce projet contribue au développement de l’agriculture numérique en fournissant des outils efficaces pour l’enrichissement des jeux de données d’imagerie aérienne. Les méthodes développées offrent un compromis optimal entre réalisme des transformations et performances computationnelles, ouvrant la voie à des applications pratiques dans le domaine de l’agriculture de précision.

Bibliographie

- [1] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*.
- [2] Buslaev, A. et al. (2020). Albumentations : Fast and flexible image augmentations. *Information*.
- [3] Zhang, C. et al. (2016). Understanding deep learning requires rethinking generalization. *arXiv*.
- [4] Ronneberger, O. et al. (2015). U-net : Convolutional networks for biomedical image segmentation. *MICCAI*.
- [5] agherazzi, S. ; Overeem, I. Models of deltaic and inner continental shelf landform evolution. *Annu. Rev. Earth Planet. Sci.* 2007, 35, 685–715. [CrossRef]
- [6] <https://www.datacamp.com/fr/tutorial/complete-guide-data-augmentation>
- [7] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). *Going deeper with convolutions*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
- [8] He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
- [9] Kingma, D. P., & Ba, J. (2014). *Adam : A method for stochastic optimization*. arXiv preprint arXiv :1412.6980.
- [10] Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv :1409.1556.
- [11] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). *Densely connected convolutional networks*. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).

.1 Annexe 1

Cette annexe présente les principaux extraits de code implémentés dans le cadre de ce projet. Le code complet est disponible sur le dépôt GitHub associé.

.2 Structure de la Classe Principale

Fichier : DataAugmentation.py

La classe principale organise toutes les fonctionnalités de data augmentation en méthodes distinctes :

```
class DataAugmentation:
    def __init__(self, dataset_path):
        self.dataset_path = Path(dataset_path)
        self.images = self.load_images_with_fix()
        self.output_dir = Path("DataAugmentation_Results")
        self.output_dir.mkdir(exist_ok=True)
        self.ROTATION_ANGLE = 25
```

.3 Méthodes Simples

.3.1 Rotation 2D

```
def apply_rotation(image, angle=25):
    h, w = image.shape[:2]
    center = (w // 2, h // 2)
    matrix = cv2.getRotationMatrix2D(center, angle, 1.0)
    rotated = cv2.warpAffine(image, matrix, (w, h))
    return rotated
```

.3.2 Ajustements Photométriques

```
def adjust_brightness_contrast(image, alpha=1.0, beta=0):  
    return cv2.convertScaleAbs(image, alpha=alpha, beta=beta)
```

.4 Méthodes Agricoles Spécifiques

.4.1 Éclairage Matinal

```
def simulate_morning_light(image):  
    hsv = cv2.cvtColor(image, cv2.COLOR_RGB2HSV).astype(np.float32)  
    hsv[:, :, 0] = np.clip(hsv[:, :, 0] * 0.85, 0, 179)  
    hsv[:, :, 1] = np.clip(hsv[:, :, 1] * 1.2, 0, 255)  
    morning_image = cv2.cvtColor(hsv.astype(np.uint8),  
                                cv2.COLOR_HSV2RGB)  
    return morning_image
```

.4.2 Effet Sécheresse

```
def simulate_drought_effect(image):  
    hsv = cv2.cvtColor(image, cv2.COLOR_RGB2HSV).astype(np.float32)  
    hsv[:, :, 1] = np.clip(hsv[:, :, 1] * 0.6, 0, 255)  
    hsv[:, :, 2] = np.clip(hsv[:, :, 2] * 1.3, 0, 255)  
    drought_image = cv2.cvtColor(hsv.astype(np.uint8),  
                                cv2.COLOR_HSV2RGB)  
    return drought_image
```

.5 Annexe 2

Méthode	Temps (ms)	CPU (%)	Mémoire (MB)
Rotation 25°	2.34 ± 0.12	15	12.5
Flip horizontal	1.56 ± 0.08	12	10.2
Luminosité +	1.23 ± 0.07	18	11.8
Motion Blur	15.67 ± 0.89	45	45.2
CLAHE	22.34 ± 1.23	65	78.9
Éclairage matinal	8.90 ± 0.45	38	28.9
Effet sécheresse	9.23 ± 0.48	40	30.1

TABLE 1 – Métriques détaillées de performance (moyenne ± écart-type)