

Apprentissage non-supervisé pour l'estimation de champs de déformation à la surface de matériaux soumis à des contraintes mécaniques

Projet Recherche - Oscar Egretéau

Sous la supervision de F.Sur, LORIA

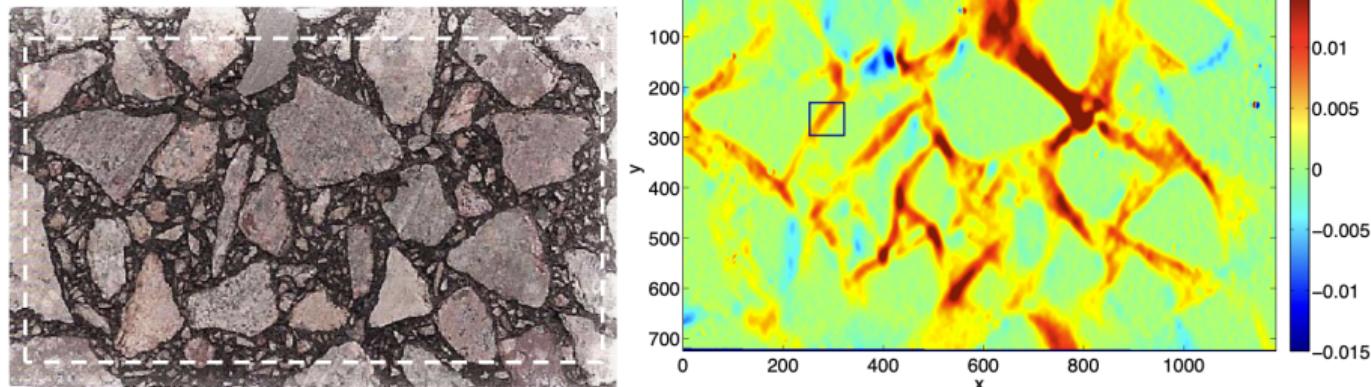
9 janvier 2026

Sommaire

1. Introduction
2. Flot Optique
 - Définition
 - Méthodes de calcul
 - Résultats
3. Réseaux de neurones
 - Définition
 - Premier exemple
 - Application au calcul du Flot Optique
4. Estimation de flot optique par apprentissage non-supervisé
5. Conclusion et perspectives

Introduction

- Analyse du flot optique pour estimer le champ de déformation de matériaux
- Apprentissage non supervisé car difficulté d'avoir une réalité terrain : pas possible de connaître le champ de déformation sur un matériau sans l'utilisation d'une méthode d'estimation numérique.



Un exemple de champ de déformation

Flot Optique

Définition

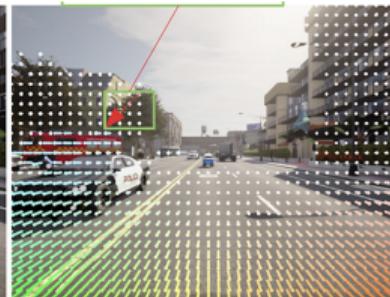
Le flot optique est un champ de vecteurs qui représente le déplacement des pixels entre deux images consécutives. Son calcul repose sur une hypothèse fondamentale, celle de la luminosité constante.



Image 1



Image 2



Optical flow

Un exemple de flot optique

Flot Optique

Définition

Le flot optique $\mathbf{u} = (u_1, u_2)$ est régi par l'équation suivante, obtenue en dérivant l'hypothèse de la luminosité constante ($\forall t \geq 0, I(t, x_1(t), x_2(t)) = I_0$) :

$$\begin{aligned}\frac{\partial I}{\partial t} + \nabla I \cdot \mathbf{u} &= \frac{\partial I}{\partial t} + \frac{\partial I}{\partial x} u_1 + \frac{\partial I}{\partial t} u_2 \\ &= 0\end{aligned}$$

Qui est un problème mal posé car les solutions $\mathbf{u} = (u_1, u_2)$ sont posées sur une droite : il en existe une infinité.

Flot Optique

Méthodes de calcul - TVL1

La méthode TV-L1 est une méthode qui se base sur la minimisation de la fonctionnelle suivante :

$$E(u) = \int_{\Omega} (|\nabla u_1| + |\nabla u_2| + \lambda |\rho(u)|)$$

Où :

- $u = (u_1, u_2)$ est le champ de flot optique.
- $|\nabla u_1| + |\nabla u_2|$ est le terme de régularisation.
- λ est le facteur de régularisation : plus λ est petit, plus le résultat sera lisse. Au contraire, pour λ grand, le résultat sera plus proche de la réalité, mais aussi possiblement plus bruité.
- ρ est le développement de Taylor de $I_1(x + u) - I_0(x)$ pour localement linéariser le problème.

Flot Optique

Premiers calculs

First Frame



Second Frame



Flot Optique

Premiers calculs

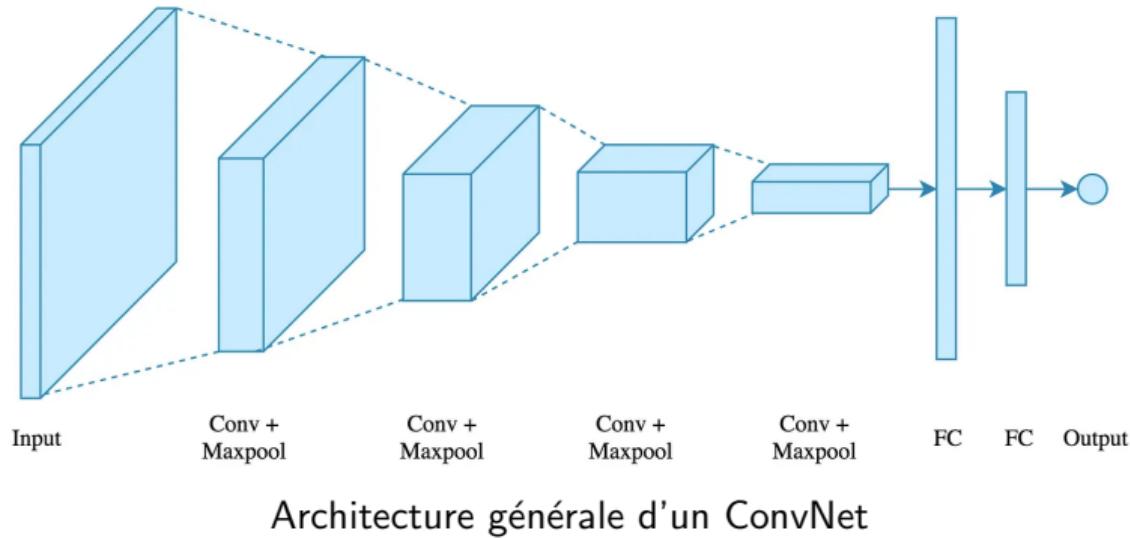
Optical Flow



→ D'où l'utilité d'utiliser les réseaux de neurones

Réseaux de neurones

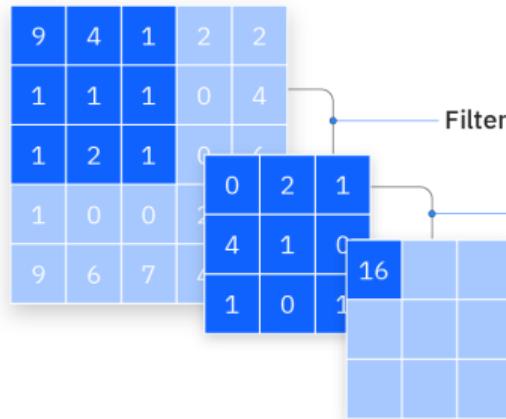
Définition - ConvNets



Réseaux de neurones

Définition - ConvNets

Input image



Output array

$$\begin{aligned} \text{Output } [0][0] &= (9 * 0) + (4 * 2) + (1 * 4) \\ &+ (1 * 1) + (1 * 0) + (1 * 1) + (2 * 0) + (1 * 1) \\ &= 0 + 8 + 1 + 4 + 1 + 0 + 1 + 0 + 1 \\ &= 16 \end{aligned}$$

Etape de convolution

Réseaux de neurones

Définition - Loss

La loss est une fonction de l'ensemble des poids affectés à chaque neurone. L'objectif de l'apprentissage est de trouver les poids qui minimisent la loss.

- Chaque type problème utilise une type de loss différente (classification,...)

Réseaux de neurones

Premier exemple

Implémentation d'un premier réseau de neurones, qui vise à classifier des fleurs selon plusieurs familles différentes :

- Marguerites
- Pisselits
- Roses
- Tournesols
- Tulipes

Réseaux de neurones



Exemples d'images dans le dataset utilisé

Le dataset contient 3500 images différentes.

Réseaux de neurones

Premier exemple

On utilise la Loss Categorical CrossEntropy, classiquement utilisée pour les classifieurs.

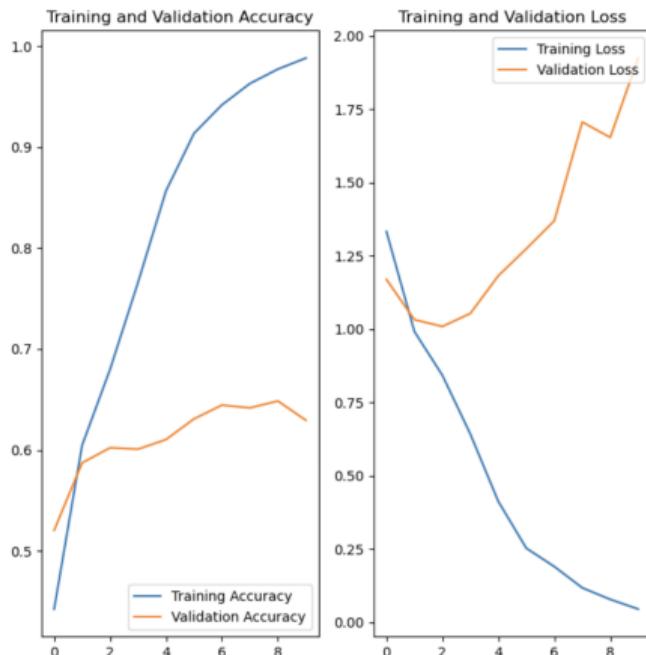
$$\mathcal{L}(p, y) = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{i,j} \log(\hat{y}_{i,j})$$

Avec :

- N est la taille de l'échantillon
- C est le nombre de classes
- $\forall(i, j) \in \llbracket 1, N \rrbracket \times \llbracket 1, C \rrbracket$, $y_{i,j} = \begin{cases} 1 & \text{Si } i \text{ est dans la classe } j \\ 0 & \text{Sinon} \end{cases}$
- $\forall(i, j) \in \llbracket 1, N \rrbracket \times \llbracket 1, C \rrbracket$, $\hat{y}_{i,j}$ est la probabilité prédite par le modèle que i soit dans la classe j .

Réseaux de neurones

Premier exemple



La loss de validation augmente avec le temps : problème du sur apprentissage

Réseaux de neurones

Correctifs au surapprentissage

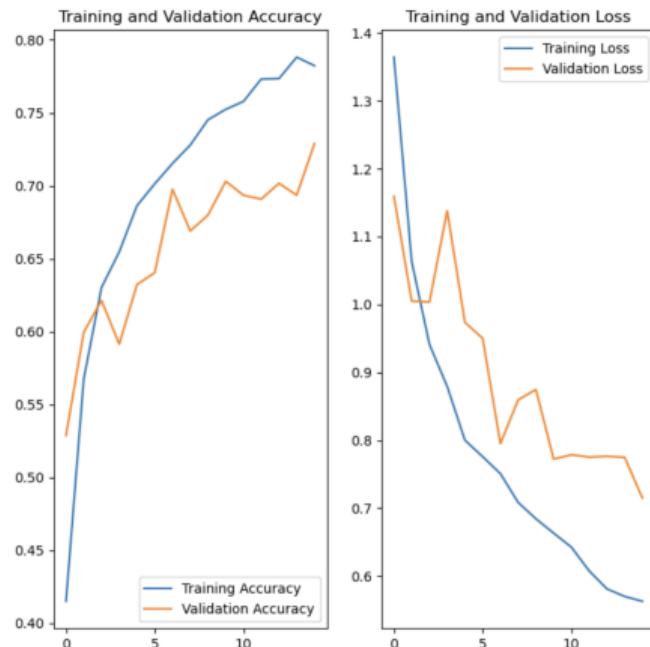
On fait de l'augmentation de data :



Réseaux de neurones

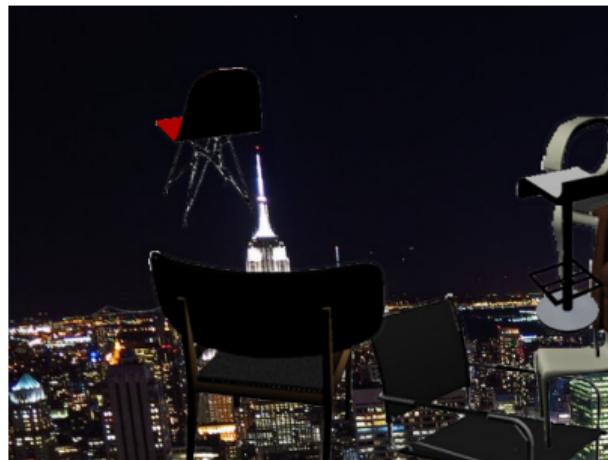
Premier exemple

Résultats



Réseaux de neurones

Application au calcul du Flot Optique



Un exemple de FlyingChairs

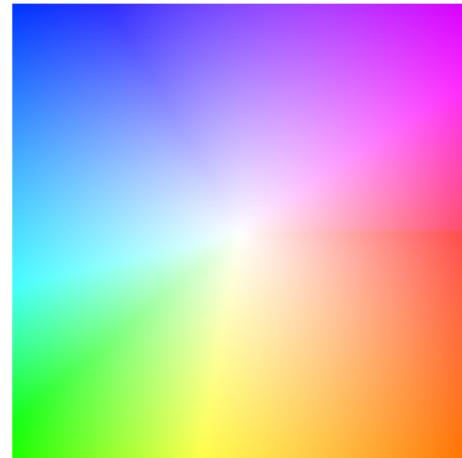
Le dataset contient 22872 images différentes.

Réseaux de neurones

Application au calcul du Flot Optique



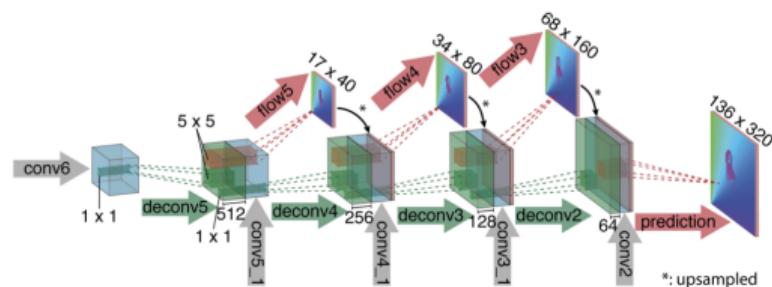
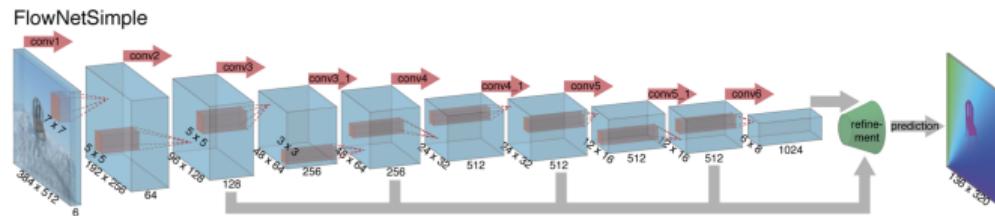
Flot optique correspondant



Roue de couleur pour lire le flot

Réseaux de neurones

Application au calcul du Flot Optique



Le modèle FlowNetSimple

Réseaux de neurones

Application au calcul du Flot Optique

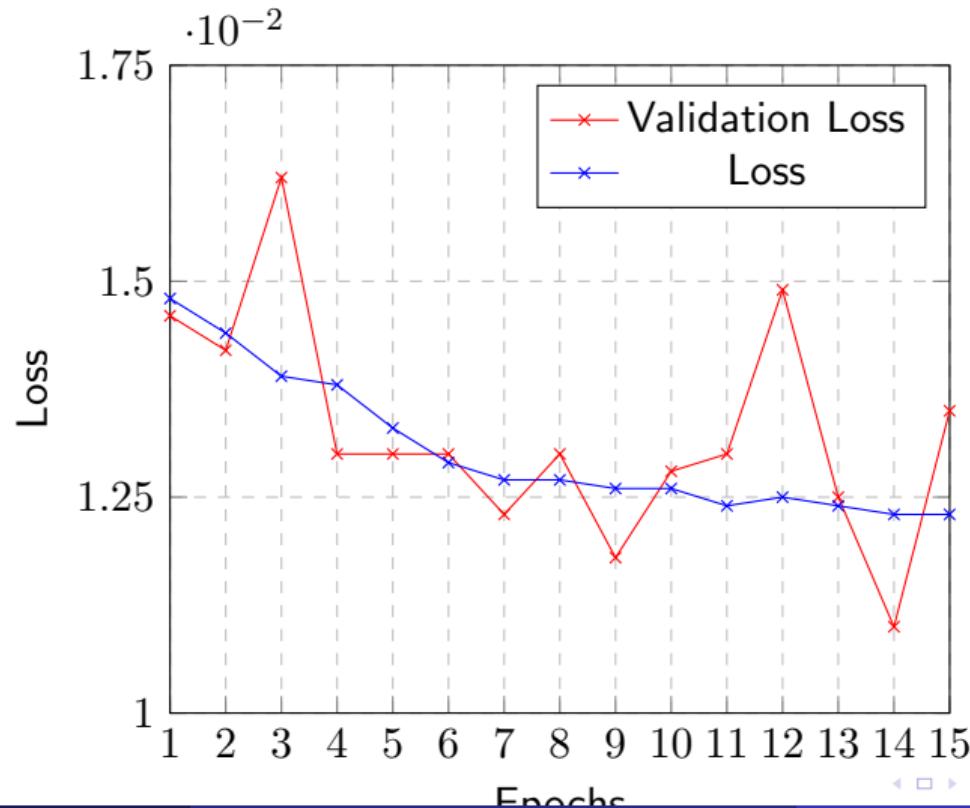
Le FlowNet utilise la loss Average End Point Error définie comme suit :

$$AEE(F, F_{\text{gt}}) = \frac{1}{HW} \sum_{i=0}^{W-1} \sum_{j=0}^{H-1} \sqrt{(u(i, j) - u_{\text{gt}}(i, j))^2 + (v(i, j) - v_{\text{gt}}(i, j))^2}$$

où $F := (u, v)$ désigne le flot optique estimé et $F_{\text{gt}} := (u_{\text{gt}}, v_{\text{gt}})$ désigne le vrai flot optique.

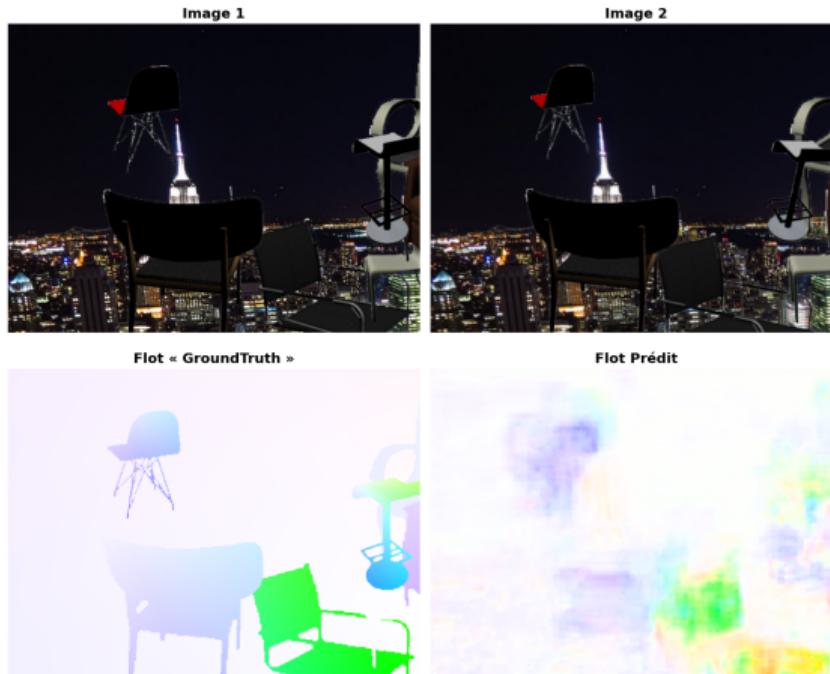
Réseaux de neurones

Application au calcul du Flot Optique



Réseaux de neurones

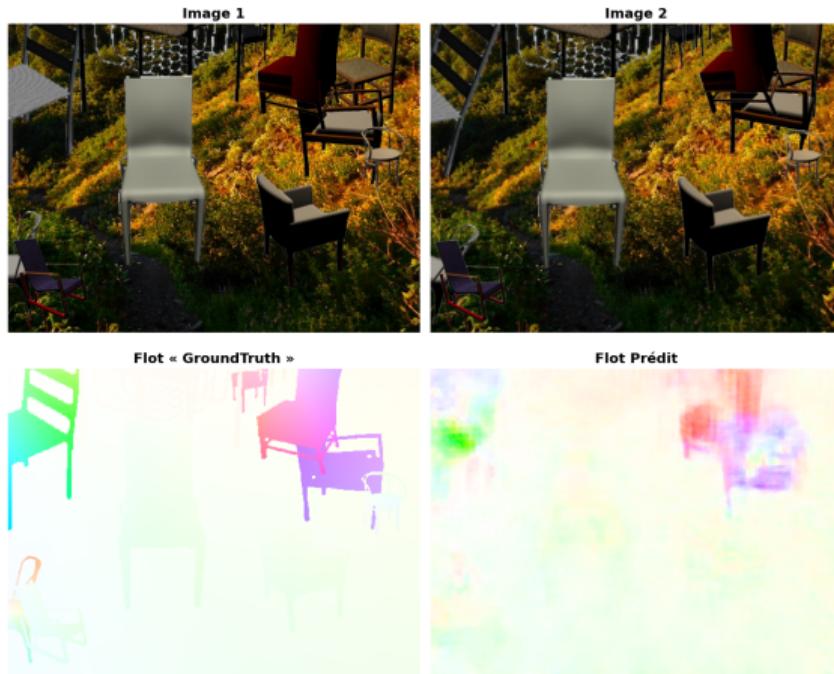
Application au calcul du Flot Optique



Résultats obtenus pour un réseau simplifié.

Réseaux de neurones

Application au calcul du Flot Optique



Résultats obtenus pour un réseau simplifié.

Estimation de flot optique par apprentissage non-supervisé

Principe : on remplace la loss AEE , par une nouvelle loss :

$$\begin{aligned}\mathcal{L}(u, v, I(x, y, t), I(x, y, t + 1)) &= \ell_{\text{photometric}}(u, v, I(x, y, t), I(x, y, t + 1)) \\ &\quad + \lambda \ell_{\text{smoothness}}(u, v)\end{aligned}$$

Où :

- $u, v \in \mathbb{R}^{H \times W}$ sont les composantes horizontales et verticales du flot prédict.
- $\ell_{\text{photometric}}(u, v, I(x, y, t), I(x, y, t + 1)) = \sum_{i,j} \rho_D(I(i, j, t) - I(i + u_{i,j}, j + v_{i,j}, t + 1))$
- $\ell_{\text{smoothness}}(u, v) = \sum_j^H \sum_i^W (\rho_S(u_{i,j} - u_{i+1,j}) + \rho_S(u_{i,j} - u_{i,j+1}) + \rho_S(v_{i,j} - v_{i+1,j}) + \rho_S(v_{i,j} - v_{i,j+1}))$

Estimation de flot optique par apprentissage non-supervisé

Où :

- $\rho_{S,D} : x \mapsto (x^2 + \varepsilon^2)^{\alpha_{S,D}}$ est la fonction de Charbonnier.
- λ est un paramètre de régularisation qui décide de l'importance relative pour le flot prédit soit lisse ou non.

On garde ensuite l'architecture du modèle FlowNetSimple.

Conclusion et perspectives

- Résultats encourageants mais encore inutilisables car trop flous.
- On s'intéresse à présent à l'apprentissage non-supervisé comme expliqué.
- Objectif : l'implémenter en Python
- L'appliquer aux matériaux