Projecte: Report

Tècniques Avançades de la Intel·ligència Artificial

2022/23

Martí Mas Fullana

Repositori:

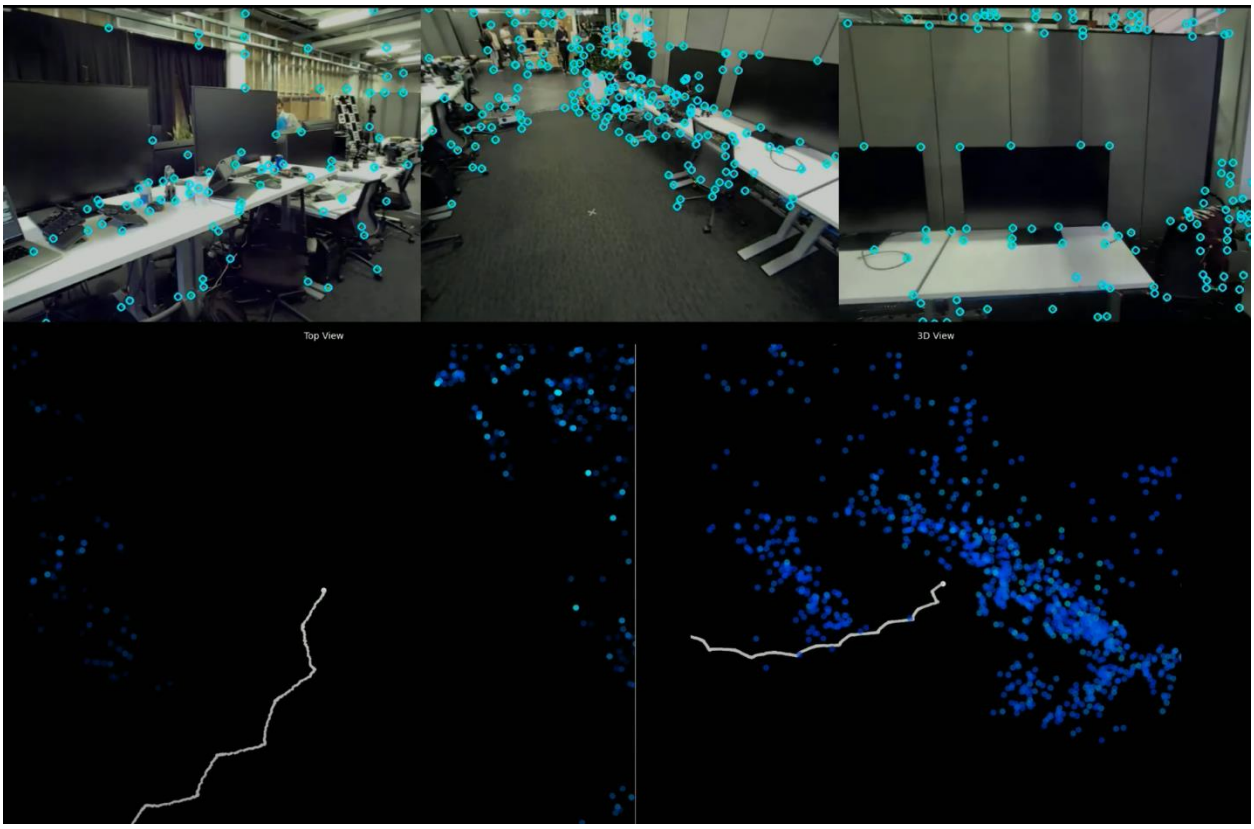
https://github.com/SupremeLobster/CfC_LiquidNetwork-DeepVO

Índex

1. Descripció del Problema

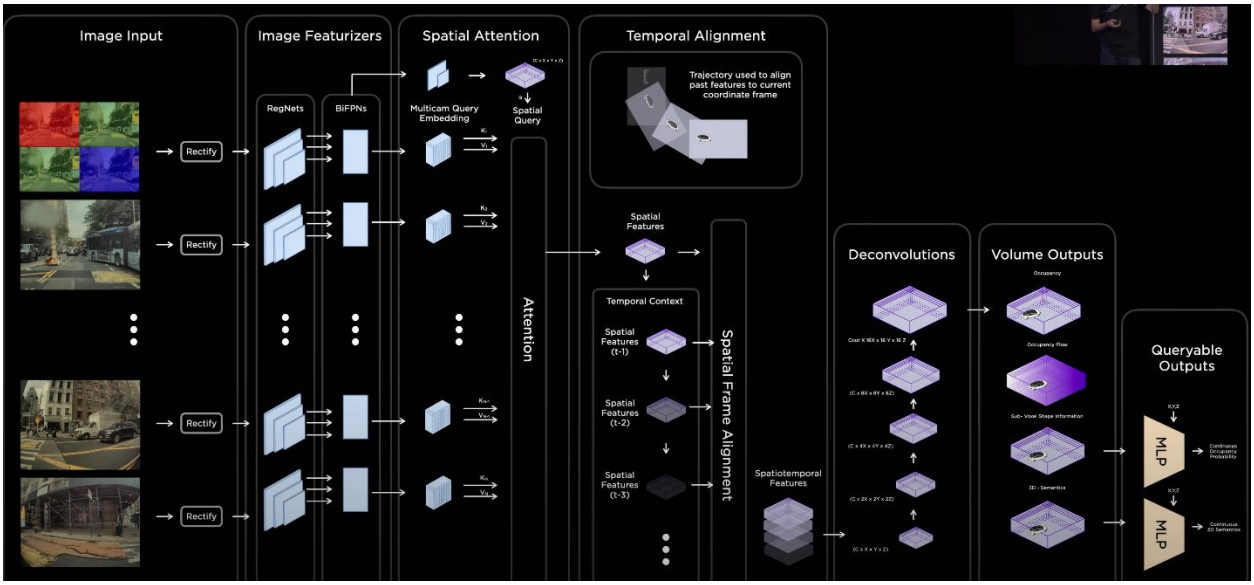
1a Part

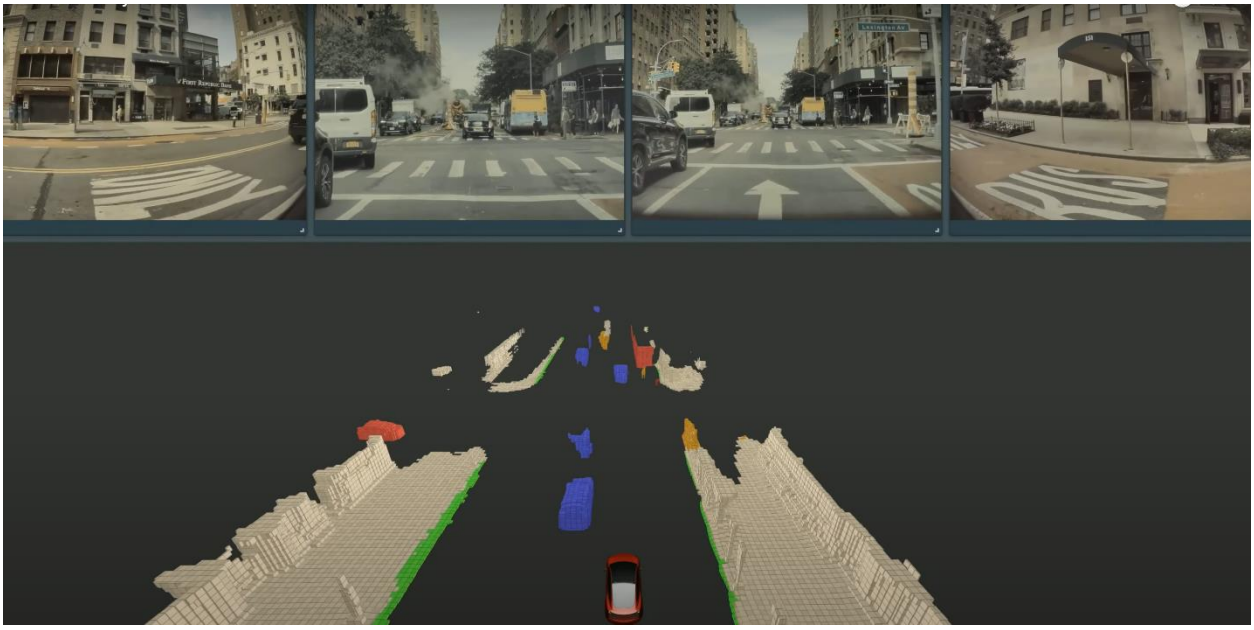
La meva proposta és crear un sistema de deep learning que estimi la posició i orientació de la càmera donat un vídeo, com es mostra a l'exemple:



2a Part (com a possibilitat)

Ja que aquest camp sembla que hi ha molta recerca feta, també proposo la possibilitat d'estendre la pràctica si la estimació de posició resultés ser relativament senzilla. Aquesta extensió consistiria en intentar replicar el sistema que va presentar Tesla a https://youtu.be/ODSJsviD_SU?t=4516





Aquest sistema prediu, a partir d'una seqüència d'imatges, la probabilitat que un voxel estigui ocupat dins d'un volum particionat a cubs (en la presencació a més ensenyen que classifiquen aquests voxels en unes 4 categories, pero jo aquest pas no el faria).

Tot això ho faré amb imatges generades amb Blender ja que això em permetria tenir moltes dades amb els valors correctes que la IA hauria de predir (blender sap la posició exacte de la càmera, per exemple) per poder fer entrenament totalment supervisat.

Per començar em centraré en la primera part i ignoraré la segona part.

2. Introducció i Motivació

Recentment la recerca en xarxes neuronals que incorporen equacions diferencials per emular el comportament de les neurones biològiques ha demostrat resultats prometedors en modelatge seqüencial (Neural ODE [1], Liquid Networks [2], CfC [3]). Amb la publicació de [3] s'aconsegueix una aproximació per la solució d'aquestes EDOs que promet incorporar escalabilitat a aquests models.

A pesar que el problema de odometria visual (1a Part) ja està molt investigat, en aquest projecte m'enfocaré a aplicar les últimes tecnologies de IA a aquest problema. Concretament l'objectiu serà incorporar les aportacions de [3] al sistema desenvolupat a [4] per aconseguir un sistema end-to-end que sigui, idealment, robust, general i explicable, al mateix temps que es respecta el principi de causalitat. Això ho faré reemplaçant el mòdul RNN de [4] per [3].

3. Dades / Coneixement del que es disposa

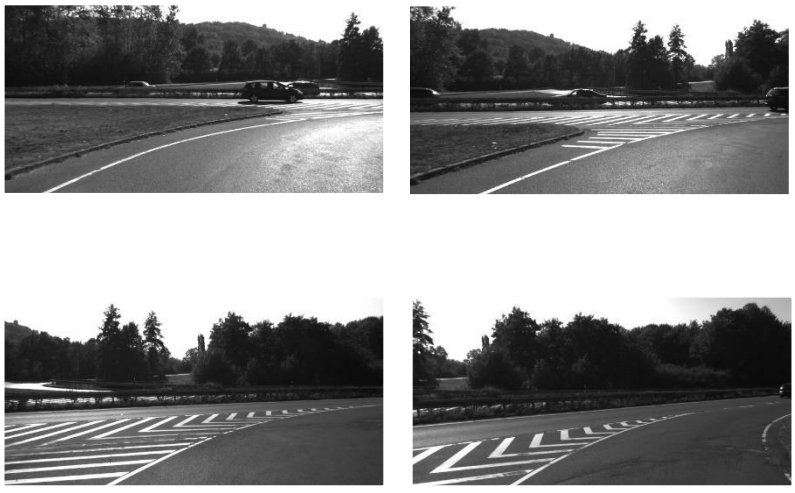
Jo personalment ja disposo d'experiència prèvia entrenant sistemes de deep learning per identificació d'objectes (localització i classificació dins la imatge) en temps real.

Per entrenar i avaluar el sistema utilitzaré el dataset de odometria visual de KITTI [5]. Concretament utilitzaré el subconjunt de clips en escala de grisos degut a limitacions de memòria i còmput. Aquest subconjunt consisteix en 22 seqüències de vídeo estereoscòpic (tot i que jo utilitzaré només 1 càmera com a input). 11 de les 22 seqüències disposen de ground truth per poder entrenar, mentre que les 11 restants no tenen ground truth i s'utilitzen per avaluar el model.

Aquest subconjunt pesa 22 GB, per tant hi ha 11 GB de vídeos estereoscòpics etiquetats per entrenar. Per tant tenim 11 vídeos que utilitzarem per entrenar i validar l'entrenament, anomenats 00 fins a 10. Utilitzaré només la càmera dreta de cada vídeo (cada vídeo consta de l'ull dret i esquerre per si es volgués fer estereoscòpia, però no és l'objectiu d'aquest projecte).

Utilitzaré la mínima quantitat de pretractament: normalitzar per tal de tenir els valors dels píxels entre 0 i 1 enlloc de 0-255. Depenent de les limitacions de hardware pot ser necessari reduir la resolució de les imatges, o fins i tot dividir un mateix clip en varies passades per mantenir l'ús de VRAM sota control.

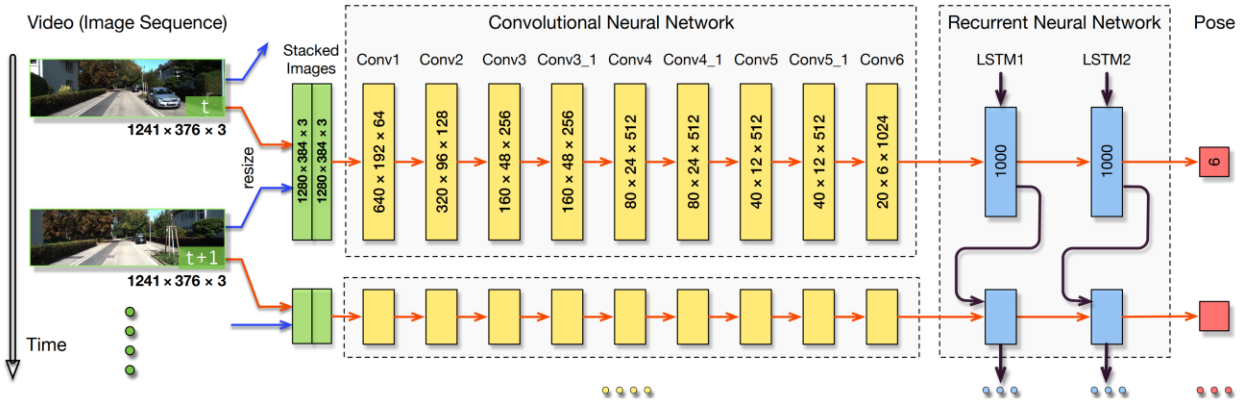
Exemple de seqüència del vídeo 01 del *dataset*



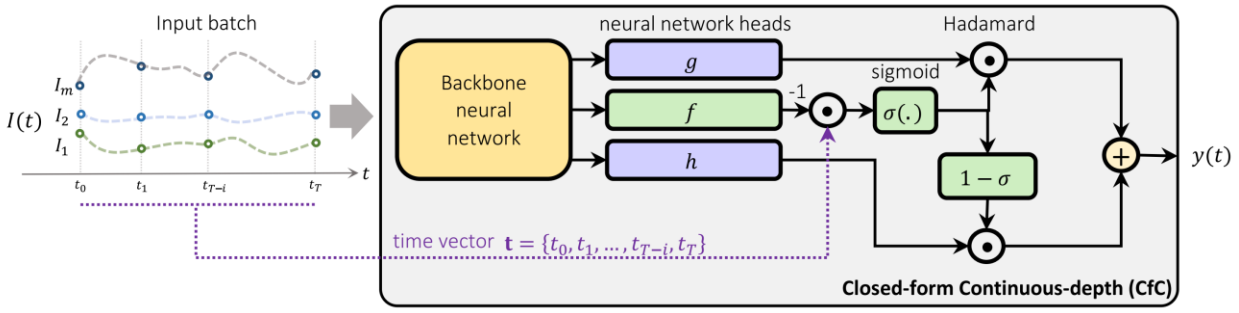
4. Proposta

L'objectiu és incorporar les aportacions de [3] al sistema desenvolupat a [4] per aconseguir un sistema end-to-end que és, idealment, robust, general i explicable, al mateix temps que es respecta el principi de causalitat. Això ho faig reemplaçant el mòdul RNN de [4] per [3].

Partint del model implementat a [4]:



L'objectiu serà reemplaçar el mòdul RNN per el mòdul de [3]:



A la pràctica això significarà que el "backbone" del segon diagrama podrà ser el CNN del primer diagrama (ja que aquest "backbone" no és part de les aportacions de [3], i per tant cal experimentar quina és la millor solució donades les restriccions de memòria, còmput i temps).

Caldrà retocar la implementació d'ambdós models per tal de donar suport a imatges en blanc i negre i poder connectar correctament les parts de les diferents xarxes (à la Frankenstein).

5. Experiments

Metodologia

El model utilitzat per al mòdul RNN és la versió LSTM de [3] (a l'article en diuen *mixed*). Bàsicament es tracta d'una LSTM convencional on es reemplacen les neurones recurrents per les de la CfC de temps-constant.

El *backbone* (la CNN que extreu *features* abans de passar-les a la RNN) és una reimplementació de FlowNet [6], però en blanc i negre (imatges d'un sol canal). No he utilitzat cap model pre-entrenat ja que els que existeixen són per imatges RGB. Tot i això, aconsegueixo resultats rellevants i prometedors.

He entrenat el model de dos maneres, amb dos conjunts de vídeos diferents del mateix *dataset*.

El model A servirà per comparar els resultats amb els de la implementació no oficial de DeepVO [7] en la que m'he basat.

El model B servirà per comparar els resultats amb els que es donen a l'article original de [4].

Per ambdós models, utilitzo els següents paràmetres per entrenar:

- Mida imatge: 304x92
- Llargades seqüències: entre 10 i 14 imatges per seqüència (amb *overlap*=1)
- *Batching*: 8 mostres per *batch*
- RNN *hidden_features*: 448
- Èpoques: 250
- Utilitzo *dropout* de 0.2 a totes les capes menys a l'última convolució i a la capa *Feed Forward* de la RNN, on utilitzo 0.5 per ambdues
- L'algoritme d'optimització és AdamW amb *learning rate* (lr) inicial de 0.002 amb un decreixement exponencial de 0.97 i un decreixement dels pesos de 0.0001 per tal d'estabilitzar l'entrenament.

L’entrenament es realitza en una sola GPU NVidia GTX 970 (4GB VRAM). La màquina utilitzada consta d’una CPU i7 4790 i 32 GB de memòria. L’entrenament dura unes 8 hores per aconseguir els resultats que s’exposen.

Es tarden uns 100 segons per època entrenant i un 15 segons per època avaluant en el conjunt de validació.

Distribució dels vídeos pel Model A:

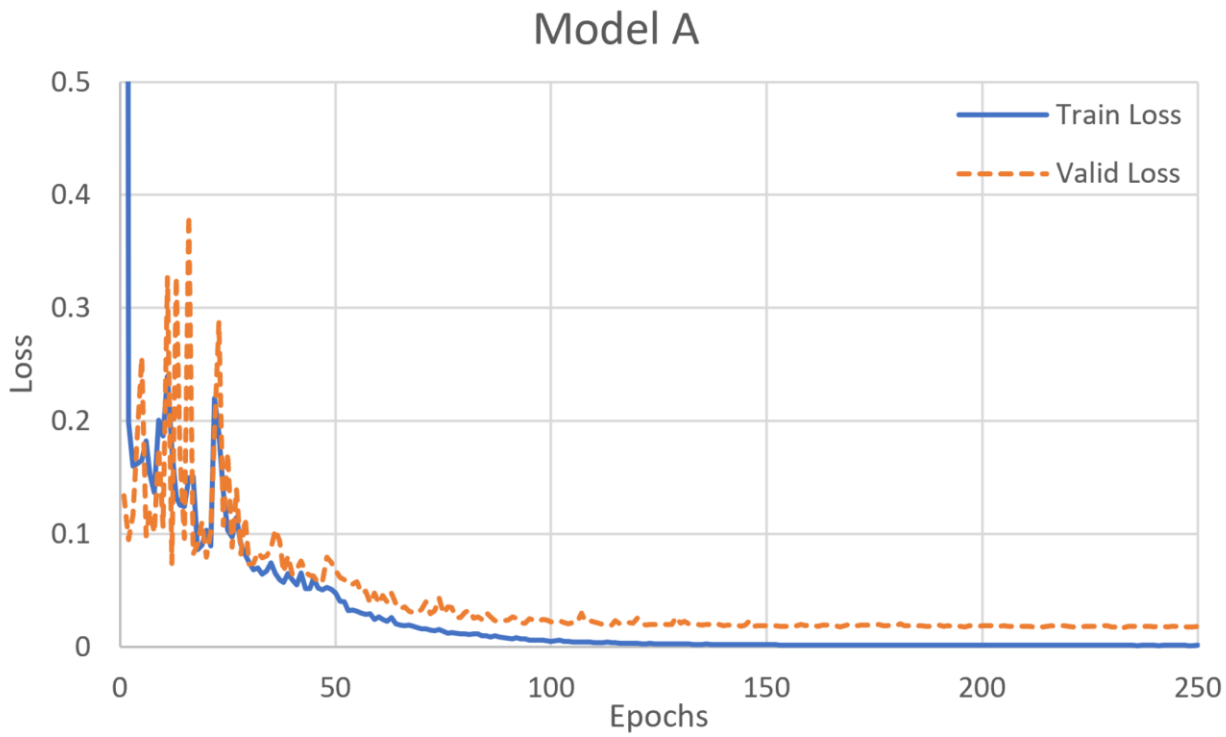
- Vídeos per entrenar: 00, 01, 02, 05, 08, 09
- Vídeos per avaluar: 04, 06, 07, 10
- Això resulta en **1697** mostres per entrenar i **330** per validar.

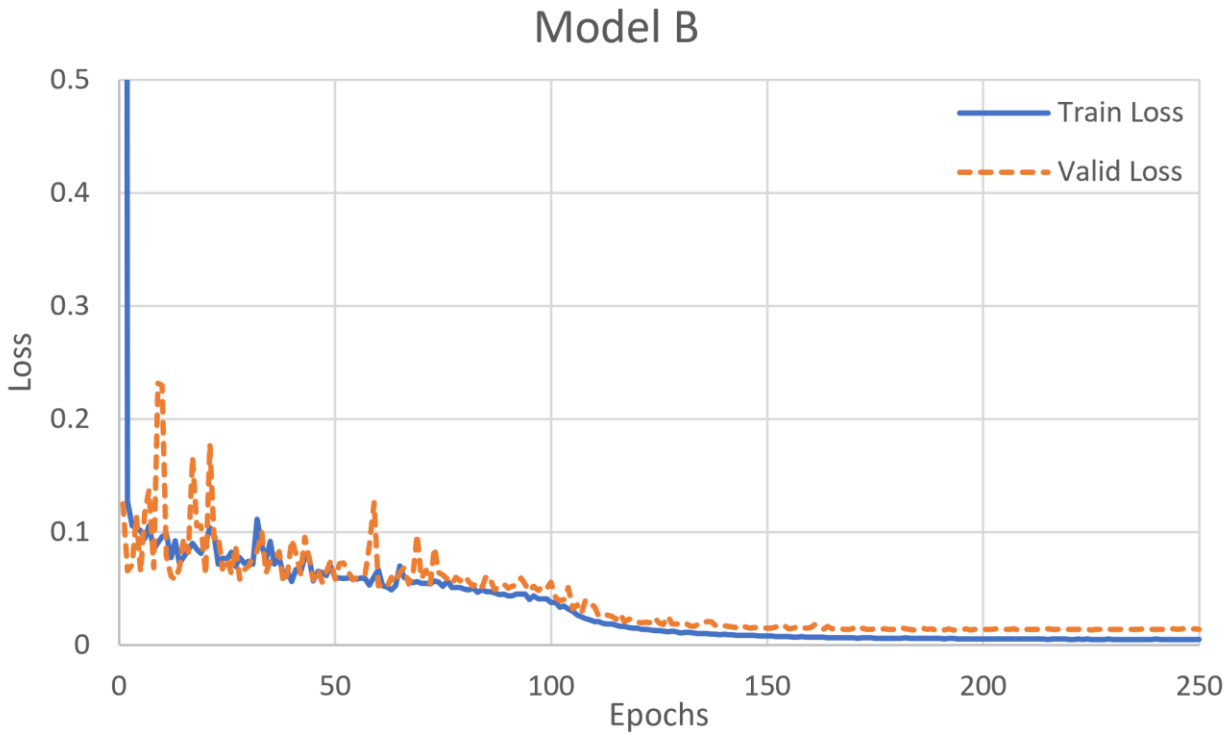
Distribució del vídeos pel Model B (corresponent a la distribució de [4]):

- Vídeos per entrenar: 00, 02, 08, 09
- Vídeos per avaluar: 03, 04, 05, 06, 07, 10
- Això resulta en **1347** mostres per entrenar i **651** per validar.

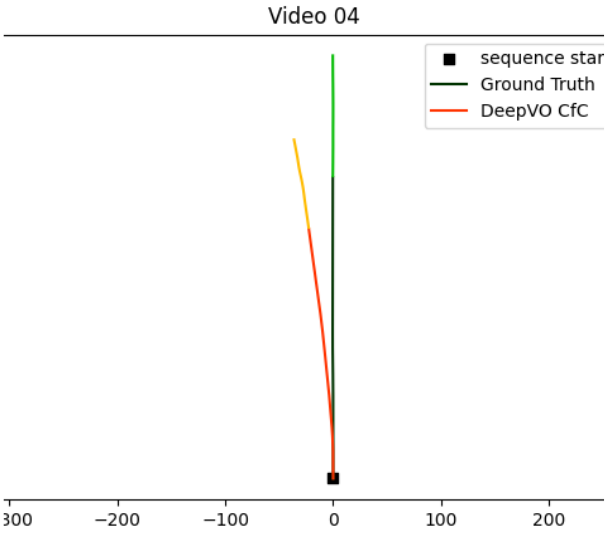
Per obtenir els resultats de la següent secció he avaluat, per cada model, la versió que ha donat el millor resultat en el conjunt de validació durant l’entrenament (*validation loss* més baix). És a dir, faig servir la tècnica de *early-stopping* per minimitzar els efectes de *overfitting*.

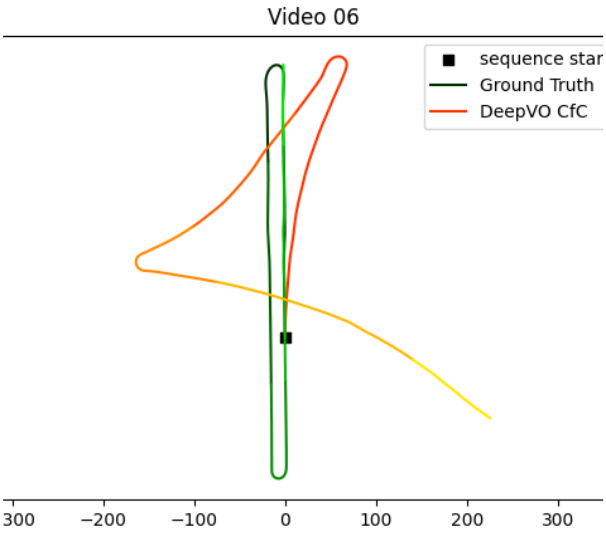
Resultats

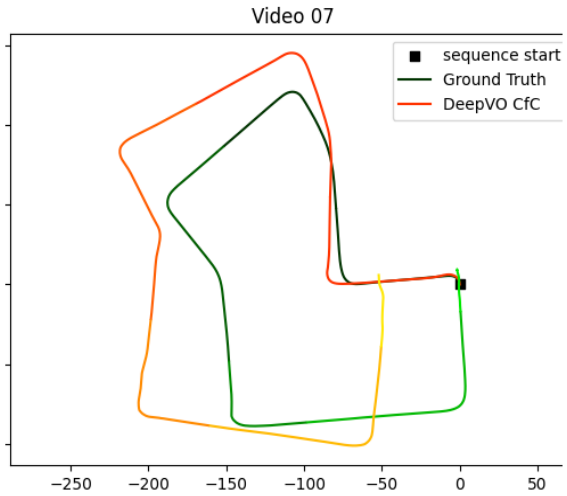

Model A
Train Loss
Valid Loss
Loss
Epochs
0.5
0.4
0.3
0.2
0.1
0
0
50
100
150
200
250

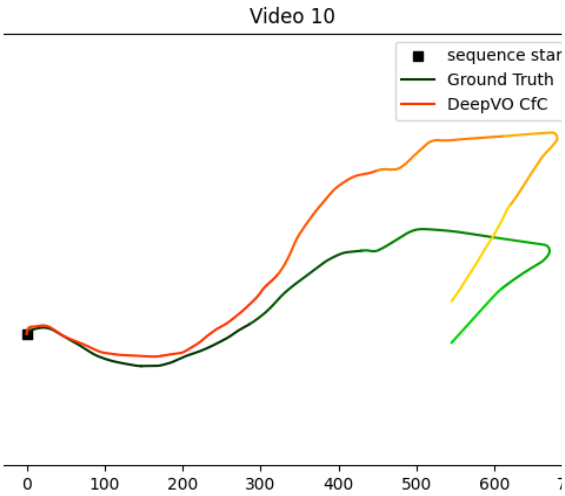

Model B
Train Loss
Valid Loss
Loss
Epochs
0.5
0.4
0.3
0.2
0.1
0
0
50
100
150
200
250

Resultats model A:

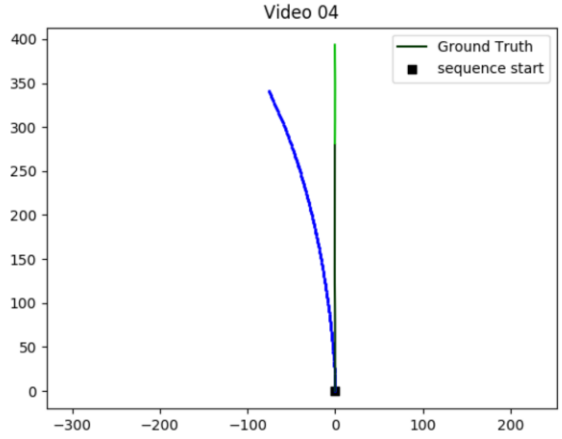


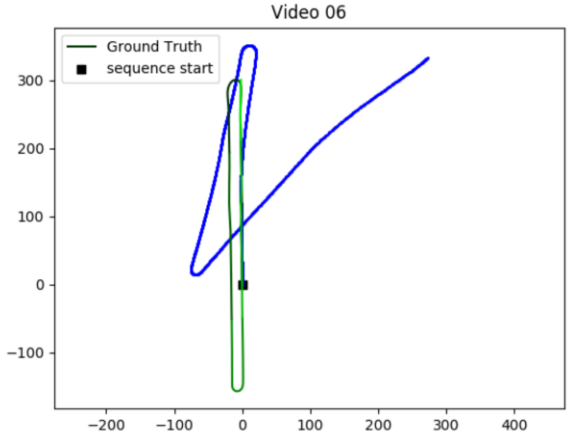


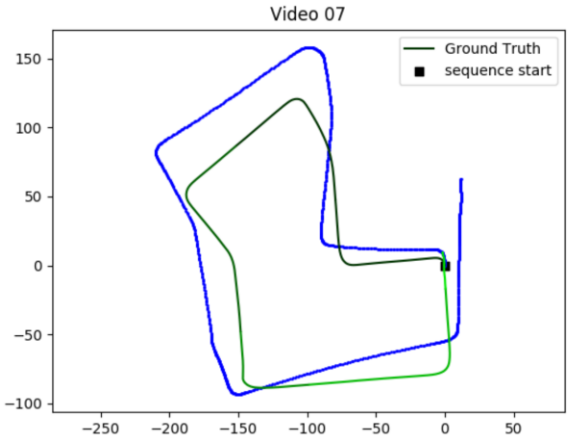


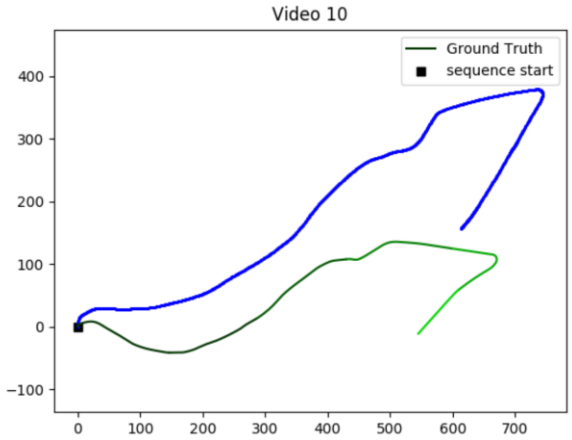


Comparació (visual) amb resultats de [7]**:**

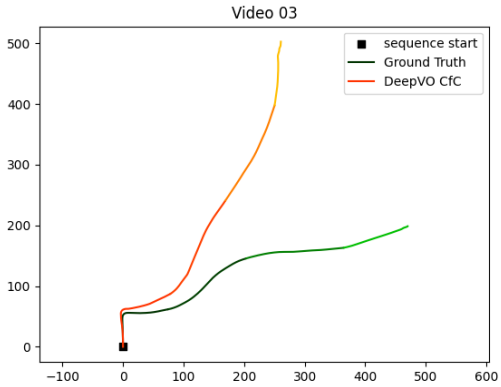


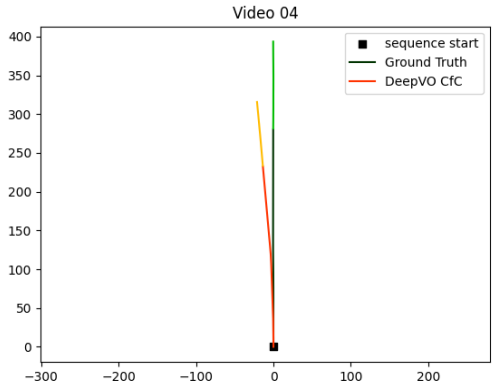


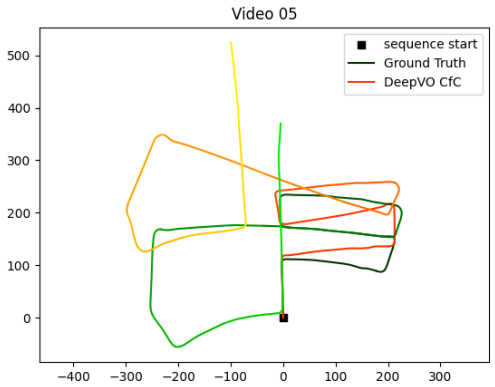


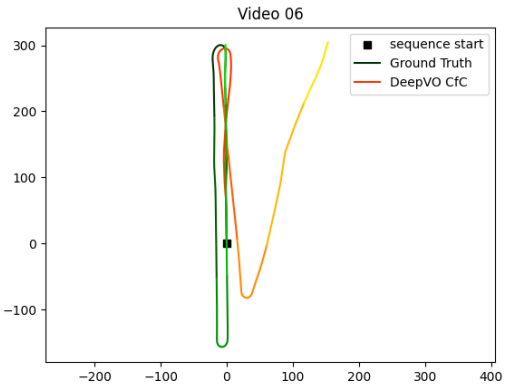


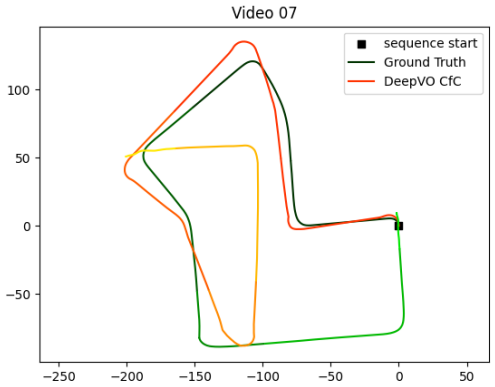
Resultats model B:

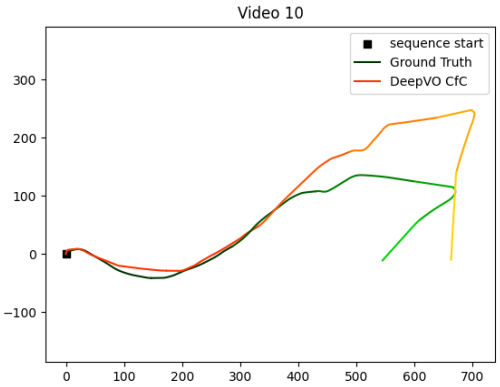






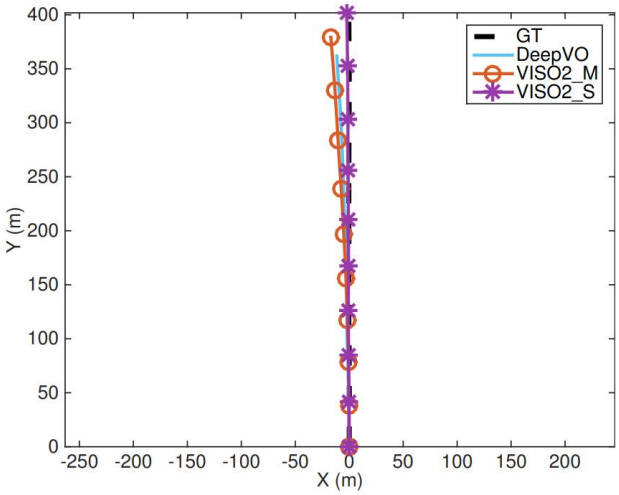




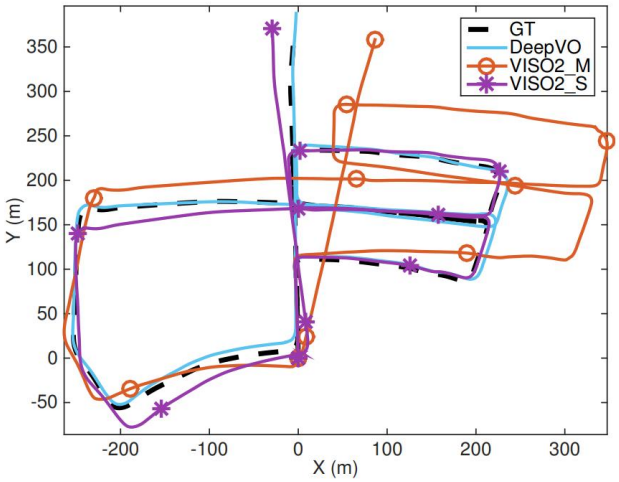


Comparació (visual) amb resultats de [4]**:**

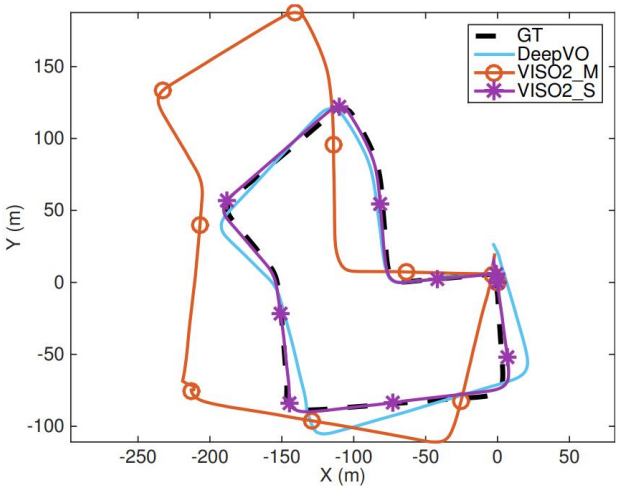
Els resultats de [4] són els que estan traçats en blau clar, etiquetats com a “DeepVO”. Els traçats de línia discontínua negra corresponen al recorregut real. Els altres traçats es poden ignorar en aquesta comparació. A [4] no donen els resultats dels vídeos 03 i 06 tot i que diuen que els utilitzen per avaluar.



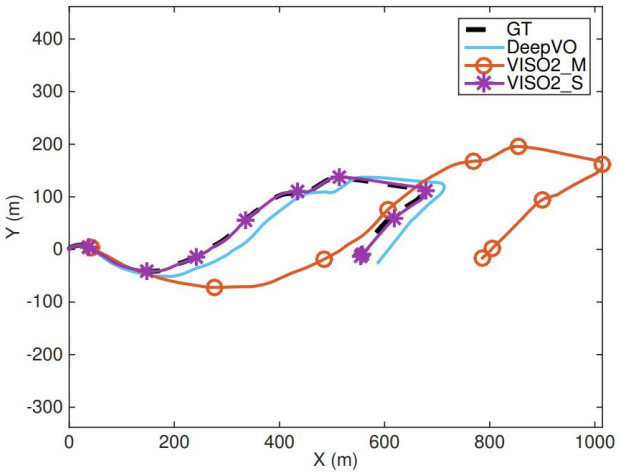
(a) Sequence 04.



(b) Sequence 05.



(c) Sequence 07.



(d) Sequence 10.

Comparació (numèrica) del model A amb el model B:

	Model A		**Model B**	
Vídeo	MSE Rotació	MSE Translació	MSE Rotació	MSE Translació
03	____		21.32	12273.71
04	**0.75**	**639.56**	**0.29**	**563.85**
05	____		26.82	4972.74
06	**299.19**	**9913.01**	**119.90**	**1657.17**
07	**185.77**	**732.26**	**56.26**	**2585.25**
10	**10.66**	**2784.84**	**21.29**	**2596.78**
Mitjana	124.09	3517.35	49.44	1850.76

Per tal de poder fer les comparacions utilitzem només els valors en negreta per calcular les mitjanes de cada model.

Podem observar que en mitjana el model B (entrenat de manera més similar a [4]) té bastant menys error tot i que ha sigut entrenat amb menys dades.

6. Conclusions i línies futures

En conclusió, malgrat haver estat entrenat amb imatges de resolució més baixa, el nou model té un rendiment relativament correcte. Això és inesperat donada la baixa resolució i suggereix que la nova tècnica és prometedora. No obstant això, per poder avaluar adequadament el seu rendiment, seria necessari entrenar el model amb imatges de la mateixa resolució que el model original.

En treballs futurs, millorar l'arquitectura del model i el procés d'entrenament també pot ajudar a millorar el seu rendiment.

En general, els resultats indiquen que el nou model té potencial per a tasques d'odometria visual, però cal fer més investigacions per aprofitar al màxim les seves capacitats.

També cal notar que a [4] i [7] utilitzen un *backbone* pre-entrenat, cosa a la que no tinc accés en aquest cas. Per tant és molt notable el fet que, amb la poca quantitat de dades i la baixa resolució d'aquestes, el model sigui capaç de donar resultats relativament adequats i comparables a (o millors que) [7]. Especulo que això és degut al fort condicionament inductiu que el CfC introdueix a la CNN, produint un model causal amb més facilitat que és capaç d'extreure *features* realment rellevants per la tasca a pesar de les dificultats exposades. En treballs futurs caldria inspeccionar la CNN resultant utilitzant tècniques d'explicabilitat (*explainability*) com *Deep Visualization Toolbox* [8].

7. Referències

[1] R. T. Q. Chen, Y. Rubanova, J. Bettencourt i D. Duvenaud, 19 June 2018. [En línia]. Available: https://arxiv.org/abs/1806.07366.

[2] R. Hasani, M. Lechner, A. Amini, D. Rus i R. Grosu, «Liquid Time-constant Networks, arXiv,» 8 June 2020. [En línia]. Available: https://arxiv.org/abs/2006.04439.

[3] R. Hasani, M. Lechner, A. Amini, L. Liebenwein, A. Ray, M. Tschaikowski, G. Teschl i D. Rus, «Closed-form continuous-time neural networks, Nature,» 15 November 2022. [En línia]. Available: https://doi.org/10.1038/s42256-022-00556-7.

[4] S. Wang, R. Clark, H. Wen i N. Trigoni, «DeepVO: Towards End-to-End Visual Odometry with Deep Recurrent Convolutional Neural Networks, arXiv,» 25 September 2017. [En línia]. Available: https://arxiv.org/abs/1709.08429.

[5] A. Geiger, P. Lenz i R. Urtasun, «Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite,» de *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[6] P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v. d. Smagt, D. Cremers i T. Brox, «FlowNet: Learning Optical Flow with Convolutional Networks, arXiv,» 26 April 2015. [En línia]. Available: https://arxiv.org/abs/1504.06852.

[7] C. W. Hsiao, «DeepVO-pytorch,» 2018. [En línia]. Available: https://github.com/ChiWeiHsiao/DeepVO-pytorch.

[8] J. Yosinski, «Deep Visualization Toolbox,» 2014. [En línia]. Available: https://github.com/yosinski/deep-visualization-toolbox.