
This is the **published version** of the bachelor thesis:

Cabanes Piqueras, Pol; Margalef Burrull, Tomás Manuel, tut. Modelización de campos de viento con Inteligencia Artificial. 2025. (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/317514>

under the terms of the  license

Modelització de camps de vent amb Intel·ligència Artificial

Pol Cabanes Piqueras

Resum– Aquest treball presenta una metodologia per predir el camp de vent utilitzant tècniques d'Intel·ligència Artificial, amb l'objectiu de reduir el temps de càlcul associat als models físics. El model proposat es basa en una xarxa neuronal U-Net condicionada amb vectors globals de velocitat i direcció, i s'entrena amb dades sintètiques generades amb WindNinja. L'estudi inclou comparatives entre funcions de pèrdua i modalitats de predicció, destacant l'eficiència del model multitasca i la seva capacitat de generalització sobre orografies no vistes.

Paraules clau– U-Net, DEM, predicció de vent, funció de pèrdua angular, xarxes neuronals, entrenament multitasca.

Abstract– This work proposes a methodology for wind field prediction using Artificial Intelligence, aiming to replace traditional physical simulations. A U-Net architecture with scalar conditioning based on wind speed and direction is trained with synthetic data from WindNinja. The model is evaluated under various loss and prediction configurations. Results show strong generalization and fast inference on unseen terrains.

Keywords– U-Net, DEM, wind prediction, angular loss, neural networks, multitask learning.

1 Introducció - Context del treball

LA PREDICCIÓ ESPACIAL DEL CAMP DE VENT és una tasca fonamental en àmbits com l'energia eòlica, la gestió de riscos meteorològics, incendis i l'agricultura de precisió. Aquests camps de vent presenten una elevada variabilitat espacial i temporal, i estan fortament influenciats per factors com la topografia i les condicions atmosfèriques locals. Fins avui, eines basades en models físics com WindNinja han estat àmpliament utilitzades per simular amb gran detall el comportament del vent sobre terrenys orogràficament complexos. Tot i la seva precisió, aquestes solucions tenen un cost computacional elevat i uns temps d'execució que poden arribar a ser d'hores o fins i tot dies, cosa que en limita la utilitat per a aplicacions en temps real o en entorns amb recursos computacionals limitats.

Aquest treball de fi de grau proposa una alternativa basada en tècniques d'aprenentatge automàtic, concretament l'ús de xarxes neuronals convolucionals (CNN) amb arquitectures de tipus U-Net condicionades. Aquestes xarxes s'entrenen per predir, a partir de models digitals d'elevació (DEM), tant la velocitat (mòdul, en km/h) com la direcció (en graus) del vent, amb una capacitat de resposta molt més ràpida que les simulacions físiques tradicionals.

L'avaluació dels resultats implica dos punts claus:

Selecció de la funció de pèrdua òptima: es comparen diverses funcions de pèrdua multitasca, combinant MSE, MAE, Huber i variants circulars per tractar la direcció angular. L'avaluació es realitza mitjançant les mètriques RMSE (per a la velocitat) i MAE circular (per a la direcció), seleccionant la que minimitza l'error global.

Comparativa de modalitats d'entrenament: amb

la funció de pèrdua escollida, s'entrenen tres configuracions diferents de xarxa:

- Predicció només de velocitat
- Predicció només de direcció, utilitzant codificació sin-cos per representar angles
- Predicció multitasca combinada, amb sortides per a velocitat i direcció

Amb aquest enfocament, s'analitza quina configuració proporciona la millor precisió amb el menor cost computacional, i es valida la seva capacitat de generalització sobre conjunts de validació i test amb orografies diverses.

2 Objectius

2.1 Objectiu general

Desenvolupar un model basat en xarxes neuronals convolucionals capaç de predir el comportament del vent —tant la seva direcció (en graus) com la seva velocitat (en km/h)— a partir de models digitals d'elevació (DEM), amb l'objectiu de complementar o substituir parcialment models físics com WindNinja. El model ha de preservar la capacitat de capturar patrons espacials complexos amb una precisió comparable a les simulacions tradicionals.

2.2 Objectius específics

Els objectius específics es poden resumir en els següents punts:

- Preparar i generar el conjunt de dades de referència (*ground truth*) mitjançant simulacions amb WindNinja. Aquest procés s'ha realitzat sobre models digitals d'elevació (DEM) en un entorn de màquina virtual proporcionat per OpenNebula (UAB), configurat específicament per permetre simulacions fi-

ables. Les sortides generades (mapes de velocitat i direcció) constitueixen la base per a l'entrenament i validació del model neuronal.

- Identificar i seleccionar la funció de pèrdua més adequada per a la regressió conjunta de la velocitat i la direcció del vent, considerant pèrdues angulars i multitasca.
- Dissenyar i entrenar diverses configuracions de xarxa neuronal, incloent:
 - Un model entrenat exclusivament per a la predicció de la velocitat.
 - Un model especialitzat en la direcció del vent (amb codificació sin-cos).
 - Un model multitasca combinat capaç de predir velocitat i direcció simultàniament.
- Avaluar el rendiment del model mitjançant mètriques com l'error quadràtic mitjà (RMSE) per a la velocitat i l'error absolut mitjà circular (MAE circular) per a la direcció.
- Refinar l'arquitectura del model mitjançant un procés iteratiu de disseny, entrenament i ajust de paràmetres, amb l'objectiu de millorar l'estabilitat i la precisió de les prediccions.
- Validar la capacitat de generalització del model sobre conjunts de validació i test que representin diferents orografies i configuracions del terreny.

3 Entorn de treball

3.1 Simulació i generació de dades

La primera fase del projecte s'ha centrat en la generació de dades sintètiques a partir de simulacions físiques realitzades amb WindNinja. Aquest procés s'ha dut a terme en una màquina virtual allotjada a la plataforma OpenNebula, proporcionada per la Universitat Autònoma de Barcelona. Aquest entorn estava específicament configurat per permetre l'execució del simulador WindNinja sobre models digitals d'elevació (DEM), obtinguts en format `.tif` amb una mida de 600×600 píxels i una resolució original de malla (*mesh resolution*) de 5 metres.

Degut a les limitacions de temps d'activitat permesa dins la màquina virtual, s'ha optat per augmentar la resolució de malla a 10 metres, fet que permet reduir significativament el temps d'execució de cada simulació. Com a conseqüència directa, les sortides generades per WindNinja (en format `.asc`) presenten un tamany reduït de 301×301 píxels. Aquestes sortides inclouen els mapes de velocitat i direcció. A més del mapa d'elevació, també es proporciona com a entrada un fitxer de configuració `.cfg` amb els valors globals de velocitat i direcció utilitzats com a condició inicial.

Aquest conjunt ha donat lloc a un total de 2.424 mostres, que constitueixen la base per entrenar i validar el model de predicció neuronal.

3.2 Entorn de desenvolupament i execució del model

Per al desenvolupament del model neuronal s'ha fet servir el llenguatge Python, juntament amb el framework PyTorch Lightning, que facilita la separació de la lògica del model, la gestió de dades i el procés d'entrenament. La codificació s'ha realitzat des de Visual Studio Code mitjançant connexió remota per SSH a una màquina de la universitat equipada amb unitats de processament gràfic (GPU).

La màquina en qüestió disposa d'una targeta gràfica NVIDIA GeForce RTX 2080 Ti amb 11 GB de memòria de vídeo i suporta CUDA 12.8. Aquesta infraestructura ha estat essencial per dur a terme els entrenaments del model amb precisió mixta de 16 bits, cosa que ha permès optimitzar tant el temps d'execució com l'ús de recursos.

3.3 Preparació i estructura del conjunt de dades

Cada mostra dins del conjunt de dades conté tres components fonamentals. En primer lloc, s'utilitza el model digital d'elevació com a entrada de la xarxa, redimensionat i interpolat fins a obtenir un mapa de 301×301 píxels. Aquests DEMs corresponen a retalls de mapes de Catalunya proporcionats del l'Institut Cartogràfic i Geològic de Catalunya (ICGC). Cada mapa es representa com un tensor amb un únic canal d'informació.

En segon lloc, es construeix un vector escalar de condicionament que integra tres valors: la velocitat global d'entrada (normalitzada respecte a un màxim de 30 km/h), i la representació sin-cos de la direcció angular també d'entrada. Aquest vector actua com a entrada addicional al model neuronal, influenciant diverses etapes de la seva arquitectura interna.

Finalment, la sortida esperada (o ground truth) està formada per un tensor amb dos canals. El primer canal representa la velocitat del vent (també normalitzada) i el segon conté el valor de la direcció del vent expressat en graus. El conjunt de dades és segmentat automàticament en tres subconjunts: entrenament (70%), validació (20%) i test (10%), utilitzant una llavor aleatòria fixa per garantir la reproduïbilitat dels resultats.

3.4 Augmentació i registre d'experiments

Amb l'objectiu de millorar la capacitat de generalització del model, s'aplica augmentació de dades durant la fase d'entrenament. Aquesta consisteix en rotacions aleatòries dels mapes d'elevació i de les sortides esperades en múltiples de 90 graus, amb l'ajust corresponent dels angles per mantenir la coherència geomètrica.

A més, s'ha incorporat un sistema de registre de mètriques mitjançant CSVLogger, que permet guardar l'evolució de la pèrdua i altres indicadors durant l'entrenament. Addicionalment, el sistema exporta al final de cada època un fitxer `.csv` amb les prediccions del model i els valors de referència per a una mostra del conjunt de validació. Aquesta informació facilita l'anàlisi detallada

dels resultats i permet visualitzar les diferències entre les prediccions i la realitat simulada.

Per entendre amb més claredat el funcionament global del sistema, a la Figura 1 es mostra el flux complet de treball implementat en aquest projecte.

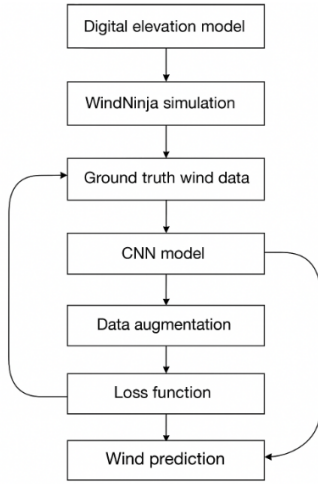


Figura 1: Flux de treball general per a la predicció de camp de vent a partir de DEMs: simulació amb WindNinja, construcció del dataset, entrenament de la xarxa i inferència final.

3.5 Flux de treball general

El sistema implementat segueix un flux de treball modular, dissenyat per transformar dades topogràfiques en prediccions precises de velocitat i direcció del vent. Aquest procés s'inicia amb la preparació del model digital d'elevació (DEM), que representa el terreny sobre el qual es vol simular el comportament del vent. Aquest DEM s'introdueix al simulador físic WindNinja, que genera, per a unes condicions globals de vent donades, els mapes corresponents a la velocitat i la direcció del flux atmosfèric.

Les dades generades es processen i organitzen per construir el dataset d'entrenament, que combina, per a cada mostra, el mapa DEM, un vector de condicionament escalar (amb la velocitat i la direcció globals codificades), i els mapes simulats de velocitat i direcció com a referència (ground truth). Aquest conjunt d'entrada i sortida és utilitzat durant l'etapa d'entrenament del model neuronal, basat en una arquitectura U-Net condicionada. Durant aquesta fase, s'aplica augmentació de dades i es calcula iterativament una funció de pèrdua ponderada que mesura la discrepància entre la sortida del model i els valors reals. L'objectiu és reduir progressivament aquesta pèrdua per tal d'aconseguir un model que generalitzi correctament a nous mapes d'elevació. La Figura 1 mostra aquest flux de manera esquemàtica, des de l'entrada geoespacial inicial fins a l'inferència final del model.

4 Estat de l'art

4.1 Models físics per a la predicció de camps de vent

Els models físics basats en equacions de la dinàmica de fluids han estat històricament els més utilitzats per predir la direcció i la velocitat del vent en entorns reals. D'entre aquests, *WindNinja* és una de les eines més esteses. Aquest simulador és capaç de calcular el flux del vent sobre terrenys orogràficament complexos mitjançant la resolució simplificada de les equacions de Navier-Stokes.

La seva utilitat rau en la capacitat de generar prediccions fiables que tenen en compte tant la topografia com les condicions globals del vent (velocitat i direcció). Tot i la seva precisió, aquests models tenen com a inconvenient un elevat cost computacional, fet que en limita l'ús en entorns amb recursos reduïts o aplicacions que requereixen respostes en temps real. En aquest projecte, WindNinja s'ha utilitzat per generar dades sintètiques que actuen com a referència de veritat (*ground truth*) per entrenar la xarxa neuronal proposada.

Taula 1: Comparativa qualitativa entre WindNinja i models neuronals

Característica	WindNinja (Model físic)	Xarxa CNN condicionada
Entrada	DEM + condicions globals (velocitat i direcció)	DEM + vector escalar (velocitat i direcció)
Sortida	Mapa de velocitat i direcció del vent	Mapa predit de velocitat i direcció del vent
Temps d'execució	Elevat (pot requerir hores per simulació)	Molt reduït (inferència en pocs segons)
Precisió	Alta, basada en models físics	Variable (depèn de l'entrenament i el dataset)
Aplicació en temps real	Limitada	Potencialment aplicable
Requisits de càlcul	Processadors potents o clústers	Inferència possible en GPU o CPU moderna

4.2 Aprenentatge automàtic i CNNs en entorns geoespacials

En els darrers anys, les xarxes neuronals convolucionals (CNN) han demostrat una gran eficàcia en tasques de regressió espacial i segmentació, essent aplicades amb èxit a problemes com la classificació d'imatges satel·litals, la detecció de cobertes terrestres o la predicció de precipitacions i temperatures.

Aquestes arquitectures són especialment útils per extreure patrons locals rellevants tot mantenint la coherència espacial de les imatges d'entrada.

L'ús de CNNs per predir camps físics com el vent encara no és tan estès com en altres àrees de la meteorologia, però representa una alternativa prometedora als models físics, especialment quan es disposa d'un gran volum de dades de referència.

En aquest projecte, s'adopta aquesta aproximació per predir tant la intensitat com la direcció del vent a partir de DEMs i condicions globals, oferint una solució híbrida entre dades geoespacial i aprenentatge profund.

4.3 Arquitectures U-Net i condicionament escalar

D'entre les arquitectures més populars per a tasques de segmentació i regressió espacial, la U-Net destaca per la seva estructura simètrica encoder-decoder i l'ús de connexions *skip*. Aquestes permeten combinar informació espacial de baixa i alta resolució, afavorint prediccions precises.

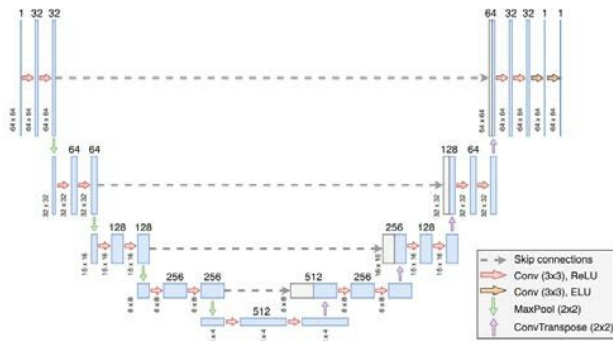


Figura 2: Arquitectura clàssica d'una U-Net per a segmentació o regressió espacial, amb estructura simètrica encoder-decoder i connexions *skip*.

Per aquest projecte s'ha fet servir una U-Net modificada amb condicionament escalar. Aquesta capacitat permet incorporar informació global (velocitat i direcció del vent) mitjançant:

- Una injecció inicial al primer bloc convolucional.
- Una fusió al *bottleneck* mitjançant una capa MLP.

Aquesta estratègia permet que el model tingui en compte tant la morfologia del terreny com les condicions globals, afavorint la generalització sobre escenaris orogràfics diversos.

Resum i motivació del projecte

Els models físics com WindNinja ofereixen una alta precisió, però amb un cost computacional elevat. Les CNNs, i en particular la U-Net condicionada, permeten una alternativa àgil i escalable per predir camps de vent, aprofitant l'eficiència de les xarxes neuronals per generalitzar sobre noves orografies.

Aquest estat de l'art justifica la necessitat d'explorar i comparar diferents configuracions i funcions de pèrdua dins una arquitectura neuronal moderna, amb l'objectiu de substituir parcialment les simulacions físiques tradicionals per sistemes d'inferència molt més ràpids i adaptables.

5 Metodologia i planificació

5.1 Planificació del projecte

La planificació del projecte ha seguit una estructura iterativa i adaptativa, dividida en sis fases temporals. La Taula 2 resumeix la planificació final, adaptada a les necessitats reals del projecte.

Taula 2: Planificació final del projecte

Setmanes	Fase	Tasques
1–3	Fase 1	Recerca i revisió bibliogràfica sobre predicció de camps de vent i xarxes CNN.
4–5	Fase 2	Generació de dades sintètiques mitjançant simulacions amb WindNinja aplicades a DEMs.
6–11	Fase 3	Desenvolupament del model: implementació de la U-Net, condicionament escalar i preparació del dataset.
12–15	Fase 4	Entrenament del model, ajustos iteratius, tuning d'hiperparàmetres i anàlisi de mètriques.
16	Fase 5	Avaluació qualitativa i quantitativa del model utilitzant el conjunt de test.
17	Fase 6	Redacció final de la memòria i preparació de l'entrega.

Durant la fase 4, es van introduir millores clau com el trasllat del vector escalar a una branca MLP dins del *decoder*, i l'afegit de convolucions post-upsampling per estabilitzar l'entrenament. Aquest cicle iteratiu de millora s'ha repetit fins a obtenir una convergència òptima.

5.2 Preparació del conjunt de dades

El conjunt de dades utilitzat s'ha generat mitjançant el simulador WindNinja, aplicat sobre models digitals d'elevació (DEM) amb característiques topogràfiques constants i una velocitat de vent fixa. L'únic paràmetre variable és la direcció del vent, que ha estat simulat des de 0° fins a 360° en intervals de 5 graus, donant lloc a 2.424 mostres totals.

Aquest conjunt s'ha dividit en tres subconjunts de manera estratificada: 70% per a entrenament, 20% per a validació i 10% per a test. Les sortides generades inclouen mapes de velocitat (en km/h) i direcció del vent (en graus), acompanyats dels valors globals utilitzats com a condició d'entrada.

5.3 Vector de condicionament escalar

Amb l'objectiu d'incorporar informació global a la xarxa, s'ha construït un vector de condicionament escalar de 3 dimensions, compost per:

- La velocitat d'entrada normalitzada respecte a un màxim de 30 km/h
- El sinus de la direcció del vent (en radians)

- El cosinus de la direcció del vent

Aquest vector permet injectar al model una representació contínua i robusta de les condicions inicials. La xarxa ha estat entrenada sota tres configuracions: només per predicció de velocitat, només per direcció, i en configuració multitasca (velocitat + direcció simultàniament).

5.4 Arquitectura del model U-Net condicionat

La xarxa emprada es basa en una arquitectura U-Net modificada, adaptada per fer predicció contínua amb condicionament escalar. L'estructura manté el patró clàssic encoder-decoder amb cinc nivells de profunditat, utilitzant convolucions de 3×3 , activació ReLU i normalització batch. Les *skip connections* entre capes simètriques permeten recuperar informació espacial de resolucions baixes.

El vector de condició s'incorpora en dues etapes:

- Al primer bloc convolucional, convertint-lo a un volum compatible i sumant-lo a l'activació inicial
- Al *bottleneck*, on es fusiona amb l'activació profunda mitjançant una capa MLP

La sortida del model té tres canals:

- Velocitat del vent (normalitzada)
- Sinus de la direcció
- Cosinus de la direcció

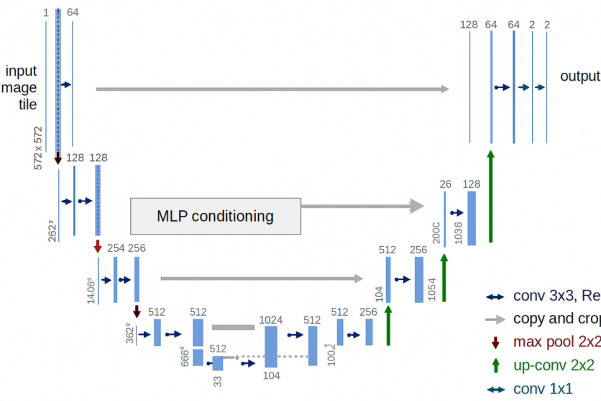


Figura 3: Arquitectura de la U-Net condicionada. El vector escalar (velocitat i direcció globals) es transforma mitjançant una MLP i s'injecta al *bottleneck* de la xarxa, permetent que les condicions globals afectin la predicció espacial.

5.5 Funció de pèrdua i mètriques

S'ha optat per una funció de pèrdua combinada, on s'utilitza:

- MSE per a la velocitat
- Smooth L1 per a l'error angular, calculat a partir de la diferència entre $\text{atan2}(\sin, \cos)$ i l'angle real en radians

La pèrdua total es pondera mitjançant la següent expressió:

$$\mathcal{L}_{\text{total}} = \lambda_v \cdot \underbrace{\text{MSE}(v_{\text{pred}}, v_{\text{gt}})}_{\text{velocitat}} + \lambda_\theta \cdot \underbrace{\text{SmoothL1}((\theta_{\text{pred}} - \theta_{\text{gt}} + \pi) \bmod 2\pi - \pi)}_{\text{error angular circular}}$$

Figura 4: Fórmula de la funció de pèrdua total utilitzada en l'entrenament.

Durant el desenvolupament, s'han comparat diverses opcions alternatives com MAE circular i Huber angular. La taula comparativa de mètriques mostra clarament que la pèrdua MSE + Smooth L1 és la més eficient:

	RMSE velocitat (km/h)	MAE angle (°)
Original (MSE+SmoothL1)	5.36	4.82
MAE-only (circular MAE)	8.13	6.28
Huber-only (circular Huber)	9.03	5.86

Figura 5: Comparació de funcions de pèrdua segons RMSE i MAE angular sobre el conjunt de validació.

A més de les mètriques finals, a la Figura 6 es mostra l'evolució de la pèrdua d'entrenament i validació per a cada tècnica de pèrdua. S'hi observa que la combinació MSE + Smooth L1 no només presenta els errors absoluts més baixos, sinó que també exhibeix una convergència més estable i ràpida que les alternatives basades en MAE o Huber angular. Aquesta estabilitat és clau per evitar sobreajustament i garantir resultats fiables.

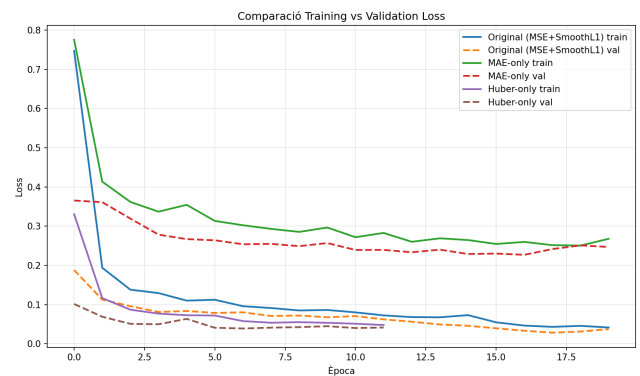


Figura 6: Comparació de l'evolució de la pèrdua durant l'entrenament i la validació per a diferents funcions de pèrdua. La variant "Original (MSE + SmoothL1)" mostra una caiguda més ràpida i sostinguda.

5.6 Entrenament i validació

Els entrenaments s'han dut a terme amb un màxim de 20 èpoques, tot i que s'ha aplicat un sistema d'early stopping amb una paciència de 5 èpoques sense millora de la pèrdua de validació. Aquest mecanisme ha permès evitar sobreajustaments en configuracions menys estables, com per exemple en el cas de la pèrdua basada en Huber, on

l'entrenament s'interromp abans d'assolir les 20 èpoques, tal com es pot observar a la Figura 6.

El model ha estat entrenat amb l'optimitzador Adam, utilitzant un *learning rate* de $5 \cdot 10^{-5}$, i un *batch size* de 8, aquest últim ajustat per evitar problemes de memòria durant l'entrenament en GPU.

El seguiment del rendiment s'ha realitzat mitjançant dues mètriques específiques:

- RMSE per a la velocitat del vent
- MAE circular per a la direcció del vent

Els millors models s'han seleccionat automàticament en funció del mínim `val_loss` obtingut durant l'entrenament. Tot i que el focus principal ha estat en la validació per optimitzar el model, també s'han generat prediccions sobre el conjunt de test per avaluar la seva capacitat de generalització.

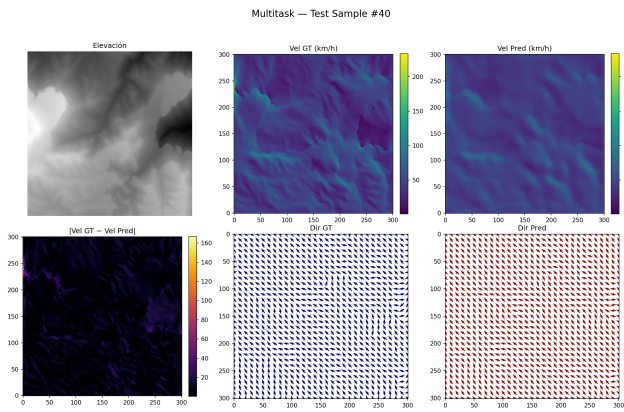


Figura 7: Exemple visual de la mostra #40 del conjunt de test. S'hi observen el DEM d'entrada, les velocitats (GT i predicció), l'error absolut i les direccions en format vectorial.

6 Resultats

6.1 Rendiment quantitatiu

Per analitzar el comportament del model entrenat, s'han avaluat les prediccions sobre els conjunts de validació i test mitjançant les mètriques RMSE (per a la velocitat del vent) i MAE circular (per a la direcció). A continuació es comparen els resultats obtinguts segons la modalitat d'entrenament emprada: model entrenat només per velocitat, només per direcció i en configuració multitasca combinada.

Els resultats mostren que la configuració multitasca ofereix un millor equilibri entre les dues prediccions, obtenint el menor MAE angular i un RMSE competitiu. Aquesta millora pot atribuir-se al fet que l'entrenament conjunt de velocitat i direcció ajuda la xarxa a aprendre representacions compartides més robustes i coherents, millorant la generalització del model. Així, es justifica l'elecció final d'una xarxa neuronal multitasca per al model definitiu.

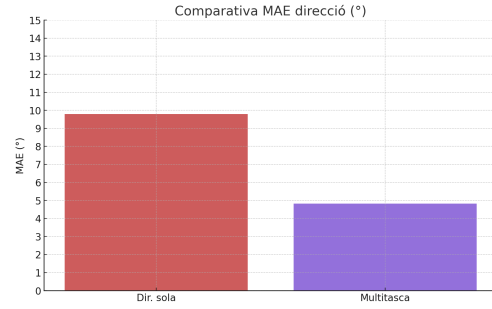


Figura 8: Comparativa d'error MAE en la predicció de la direcció per models entrenats només amb direcció (esquerra) i amb configuració multitasca (dreta).

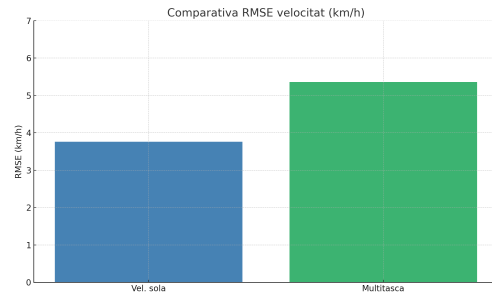


Figura 9: Comparativa de RMSE en la predicció de velocitat per a models entrenats amb velocitat sola i en configuració multitasca, en unitats km/h.

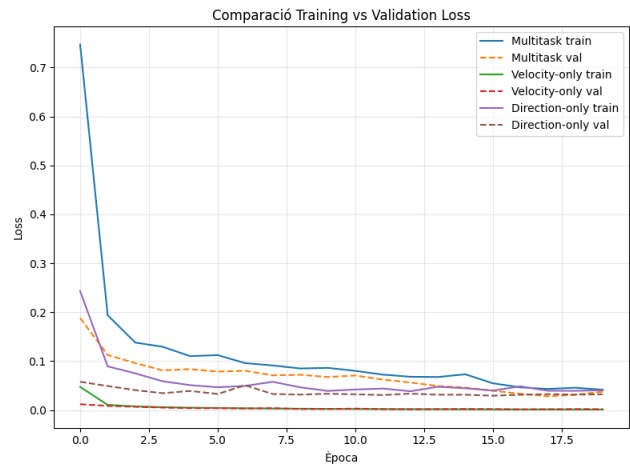


Figura 10: Evolució de la pèrdua d'entrenament i validació per a cada modalitat. La configuració multitasca presenta una convergència estable.

6.2 Visualització qualitativa

Per completar l'anàlisi, es presenten dues mostres representatives: una del conjunt de validació (mostra núm. 340) i una altra del conjunt de test (mostra núm. 75). Aquestes mostres no han estat utilitzades durant l'entrenament, fet que les converteix en una bona referència per avaluar la capacitat de generalització del model.

En ambdós casos es mostra:

- El model digital d'elevació
- La velocitat real i predita

- L'error absolut en velocitat
- Les direccions real i predita en format vectorial

Els resultats visuals posen de manifest que la xarxa és capaç de capturar patrons espacials amb un alt nivell de precisió, fins i tot en DEMs no vistos prèviament. A la Figura 10 es pot observar també com la pèrdua total s'estabilitza millor en el cas del model multitasca, indicant un aprenentatge més consistent entre les dues prediccions.

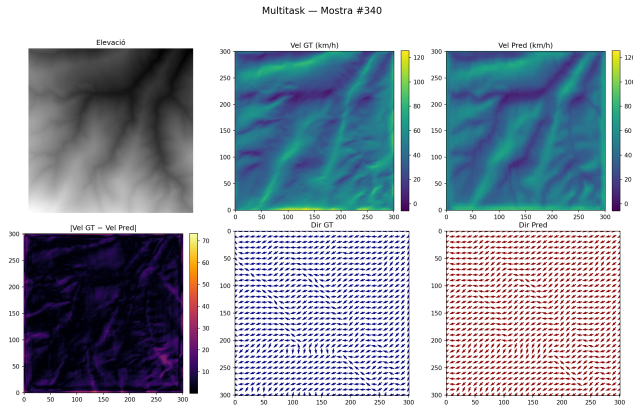


Figura 11: Exemple de la mostra #340 del conjunt de validació. El model reproduïx els patrons espacials amb elevada fidelitat.

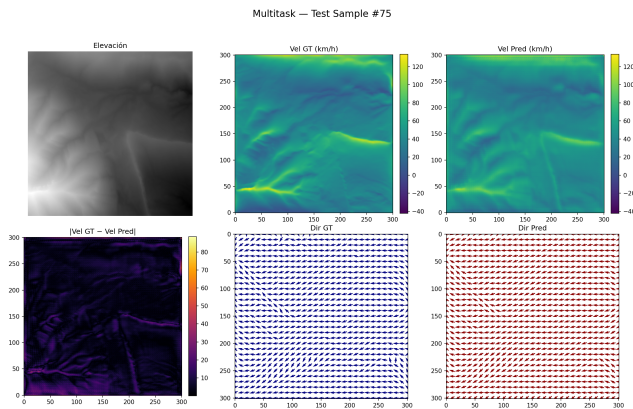


Figura 12: Exemple de la mostra #75 del conjunt de test. Es valida la capacitat del model per generalitzar sobre DEMs no vistos.

6.3 Limitacions i consideracions finals

Tot i els bons resultats obtinguts, cal tenir en compte algunes limitacions del model desenvolupat:

- **Dades sintètiques:** Totes les mostres emprades provenen de simulacions amb WindNinja. Això pot limitar la capacitat del model per generalitzar a dades reals o condicions atmosfèriques més complexes.
- **Resolució limitada del DEM:** Els models digitals d'elevació utilitzats tenen una resolució de 10 metres, cosa que pot no capturar detalls rellevants en entorns molt abruptes o urbans.

- **Condicions de velocitat fixes:** Tot i que s'ha generat un rang ampli de direccions (de 0° a 360° , en intervals de 5°), la velocitat del vent ha estat fixada per a totes les simulacions. Això redueix la variabilitat del conjunt de dades i pot limitar la capacitat del model per adaptar-se a escenaris amb intensitats fluctuants.

Aquestes limitacions obren la porta a futures millores com l'ús de dades observacionals reals, la incorporació de condicions temporals o l'experimentació amb DEMs de resolució més fina.

7 Conclusions

El present treball ha demostrat la viabilitat de substituir parcialment les simulacions físiques del camp de vent —habitualment realitzades amb eines com WindNinja— per un model neuronal entrenat específicament amb dades sintètiques.

Tot i que el simulador físic proporciona resultats altament precisos, la seva execució requereix diversos minuts o hores per escenari, mentre que el model U-Net condicionat proposat pot realitzar inferències en qüestió de segons amb una precisió comparable. L'ús del vector escalar com a condició d'entrada ha permès que el model generalitzi correctament, fins i tot sobre DEMs no vistos durant l'entrenament.

La selecció acurada de la funció de pèrdua ha estat clau: la combinació de MSE per a la velocitat i Smooth L1 per a l'error angular ha mostrat una convergència més estable i ràpida que les alternatives basades en MAE o Huber. Això s'ha reflectit en els resultats quantitatius (RMSE i MAE circular) i en la qualitat visual de les prediccions.

A més, l'entrenament multitasca ha demostrat ser l'enfocament més eficient, ja que la xarxa aprèn representacions compartides més robustes quan prediu velocitat i direcció simultàniament. Aquesta configuració ha superat en rendiment les modalitats individuals, tant en precisió com en estabilitat d'entrenament.

Com a conclusió final, aquest TFG confirma que és possible entrenar xarxes convolucionals eficients per simular fenòmens físics complexos amb costos computacionals molt reduïts. Això obre la porta a aplicacions en temps real, predicció meteorològica accelerada i integració en sistemes encastats o amb recursos limitats.

Aquest projecte es podria ampliar en el futur amb les següents línies de treball: augment de la resolució espacial dels DEMs, inclusió d'altres variables meteorològiques (com humitat o temperatura), o entrenament amb dades reals provinents d'estacions meteorològiques o LIDAR.

8 Treballs futurs

Tot i els bons resultats obtinguts, aquest projecte pot servir com a base per a diverses línies de recerca i desenvolupament futurs. Algunes extensions rellevants podrien incloure:

- **Entrenament amb dades reals:** Tot i que seria molt útil validar el model amb mesures observacionals, la instal·lació de sensors meteorològics amb

resolució espacial comparable a la del model (per exemple, cada 5 metres) és altament costosa i poc realista. Per tant, una línia futura més factible seria combinar simulacions amb reanàlisis o mesures puntuals agregades a escala més grossa per ajustar o calibrar el model.

- **Ampliació de condicions meteorològiques:** En aquest treball només s’ha considerat una velocitat fixa de vent. L’entrenament amb múltiples intensitats de vent milloraria la versatilitat i aplicabilitat del model en escenaris reals.
- **Millora de la resolució espacial:** En aquest projecte, per limitacions computacionals, es va optar per una resolució de malla de 10 metres a les simulacions de WindNinja, reduint el detall dels mapes generats. Una millora futura seria utilitzar una resolució inferior (per exemple, 5 metres) per capturar millor petites variacions del relleu que poden tenir un impacte significatiu en el flux del vent.
- **Adaptació a altres variables:** Aquesta arquitectura es podria adaptar per predir altres variables meteorològiques, com la humitat, la temperatura o la presència de boira, aprofitant la mateixa estructura amb modificacions a la sortida.
- **Implementació optimitzada per dispositius embeguts:** Un model lleuger optimitzat amb TensorRT, TFLite o ONNX podria ser integrat en sensors remots o drons per predicció en temps real.

Agraïments

Vull expressar el meu sincer agraïment al meu tutor Tomás Margalef pel seu suport constant i la seva orientació durant tot el desenvolupament del treball. Gràcies també als companys que han col·laborat o compartit idees durant el procés, així com al departament d’enginyeria de la UAB per facilitar l’accés als recursos computacionals necessaris. Finalment, agraeixo a la meva família el seu suport incondicional durant aquesta etapa final d’estudis.

Bibliografia

1. M. Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015. Disponible a: <https://www.tensorflow.org>
2. D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv preprint arXiv:1412.6980, 2014.
3. O. Ronneberger, P. Fischer, and T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, en MICCAI, 2015.
4. J. Forthofer, B. Butler, and N. Wagenbrenner, *A comparison of three approaches for simulating diurnally driven slope winds*, Atmospheric Environment, vol. 45, pp. 428–438, 2011.
5. WindNinja Simulator. USDA Forest Service, Missoula Fire Sciences Laboratory. Disponible a: <https://www.firelab.org/project/windninja>
6. PyTorch Lightning. Lightning AI. Disponible a: <https://lightning.ai>

Apèndix

A1. Composició del dataset

Cada mostra del conjunt s'ha generat mitjançant WindNinja aplicat sobre un DEM constant, modificant només la direcció del vent global (de 0° a 360°, en salts de 5°). En total es van obtenir 2.424 mostres, cadascuna amb:

- DEM d'entrada en resolució 600×600 píxels
- Velocitat i direcció globals configurades al fitxer `.cfg`
- Mapes de sortida (`.asc`) per velocitat i direcció
- Assignació automàtica a entrenament, validació o test

A2. Configuració tècnica del sistema

L'entrenament s'ha realitzat en un entorn GPU remot amb la següent configuració:

- NVIDIA RTX 2080 Ti amb 11GB VRAM
- Python 3.10 + PyTorch Lightning
- CUDA 12.8 + suport mixed precision (fp16)
- SSH via Visual Studio Code

Aquesta infraestructura ha permès optimitzar el temps d'entrenament i l'ús de memòria.