

## GeoGuessr AI

### ResNet-50

A ResNet-50 (Residual Network) egy mély konvolúciós neurális hálózat, melyet széles körben alkalmaznak különböző képfelismerési feladatokban, beleértve a geolokációs alkalmazásokat is. A ResNet-50 jelentősége a reziduális tanulási blokkokban rejlik, amelyek lehetővé teszik a hálózat számára, hogy mély rétegekben is hatékonyan tanuljon, ezzel csökkentve a gradiens eltűnésének problémáját. Ezáltal a hálózat képes komplex mintázatok felismerésére és magas szintű jellemzők kinyerésére képekből. (Computer Vision Project, 2024)

A geolokációs alkalmazásokban a ResNet-50-et például az autonóm járművek navigációs rendszereiben használják, ahol a hálózat képes azonosítani és lokalizálni különböző tereptárgyakat és környezeti elemeket a képeken. Emellett használják a városi környezetekben történő objektumfelismerésre, amelyek segíthetnek a pontosabb és megbízhatóbb geolokációs szolgáltatások nyújtásában (Zang et al., 2023).

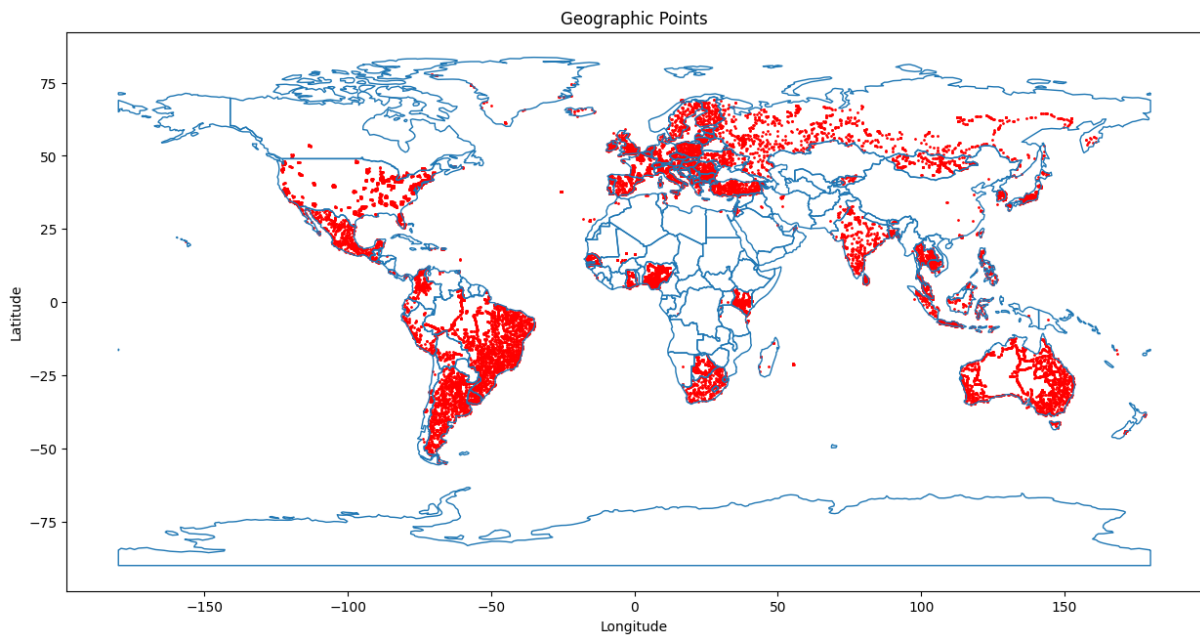
A modell további előnyei közé tartozik a transfer learning lehetősége, amely lehetővé teszi, hogy előre betanított modelleket alkalmazzunk specifikus feladatokra. Ez különösen hasznos lehet olyan ipari alkalmazásokban, mint például az acélfelületi hibák felismerése, ahol a modell átképezhető a meglévő nagy adatbázisokon, így javítva a felismerési pontosságot és csökkentve a képzési időt (Mohan, 2023).

Ilyen előre betanított modell a projektben is használt ImageNet. Az ImageNet egy hatalmas, nyilvánosan elérhető adatbázis, amely több mint 14 millió képet tartalmaz, 1000 kategóriára bontva. Az ImageNet datasetet azzal a céllal hozták létre, hogy támogassa a nagy léptékű vizuális felismerési feladatok kutatását és fejlesztését. (Deng et al., 2009)

Az ImageNet képzési adatain való tanulás segít a modelleknek, hogy jobban kezeljék a valós környezetek változatosságát és komplexitását. A különböző textúrák, anyagok és fényviszonyok, amelyek az ImageNet képein megtalálhatók, felkészítik a modelleket a valós világban történő alkalmazásokra, mint például a földrajzi helyzet felismerése különböző időjárási körülmények között

## Adat

A modell tanításához egy Kaggle-ös adatbázist használok, a [Streetview Image Dataset](#)-et. Az adat valamivel több mint 25 000 képet tartalmaz és egy külön .csv fájlban szerepelnek az egyes képekhez tartozó koordináták. A képeket a Google Streetview API-t használva gyűjtötték össze, 640x640-es méretűek.

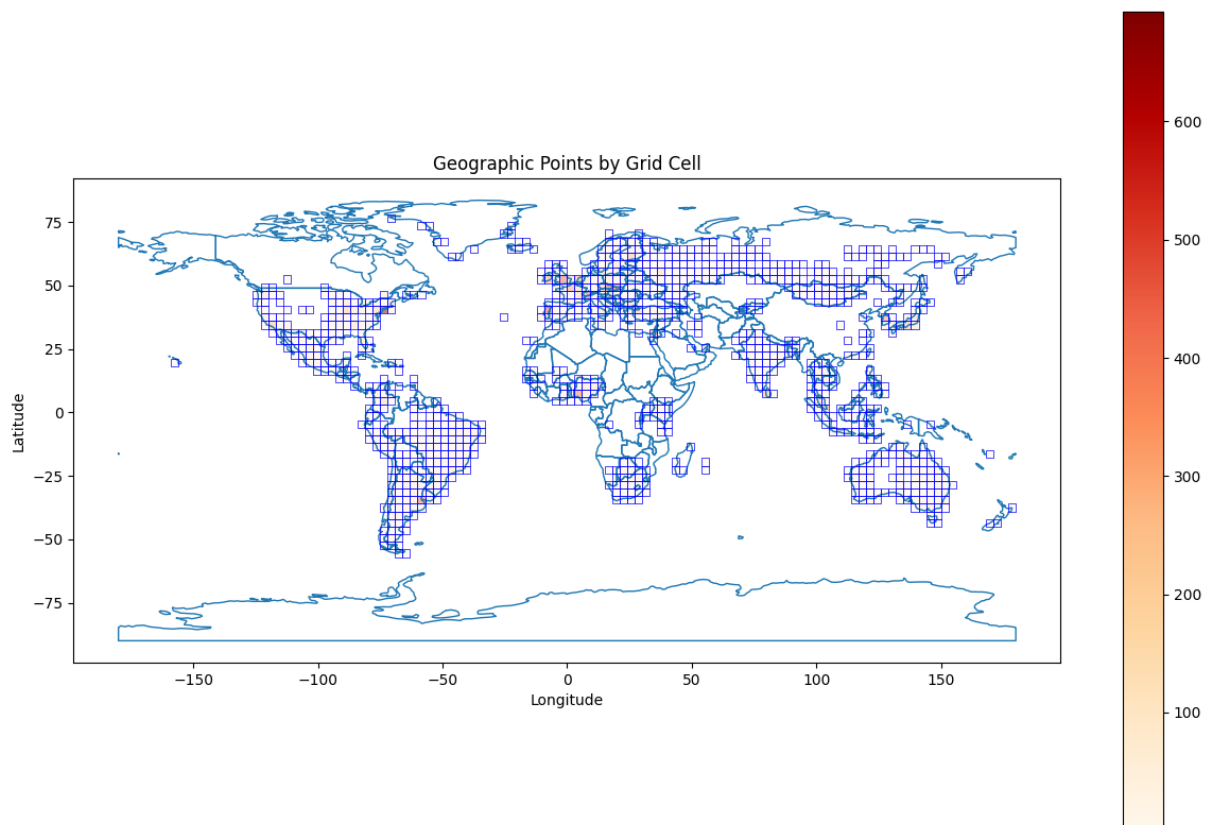


*1. ábra - Képek megoszlása (saját készítés)*

A képek nem oszlanak el egyenletesen a világtérképen, aminek legfőbb oka, hogy a Google Street View számos országban nem elérhető, vagy a rossz infrastrukturális adottságok miatt alacsony lefedettséggel bír. Az első próbálkozások során konkrét földrajzi koordinátákat akartam prediktálni a modellel, de ez egyáltalán nem bizonyult sikeresnek, ami több okból is lehet:

- A földrajzi koordináták prediktálása nagy predikációs teret jelent, amely megnehezítheti a modell számára a pontos helymeghatározást, különösen kevésbé ismert helyeken.
- A koordináták közvetlen prediktálása nehezen értelmezhető lehet, különösen a modell számára, ami befolyásolhatja a teljesítményt.
- A modell számára nehezebb lehet megtanulni az összes földrajzi hely különbözőségeit, ami a tanulási folyamatot hosszabbá és bonyolultabbá teheti.
- Lehet csak sokkal több adatot kellene használni hozzá.

Emiatt a következő próbálkozásomban már rácsokra osztottam a világtérképet, ezzel csökkentve a predikciós tér méretét, ami a tanulást és a pontos predikciót is megkönnyítheti. Emellett ez a megközelítés jelentősen egyszerűsíti a modell működését, hiszen csak egy rácsot kell prediktálni, nem pedig folytonos koordinátákat. Miután a rácsokhoz rendeltem a képeket, kitöröltem azokat a rácsokat, amik 0 megfigyelést tartalmaznak, az így kapott predikciós tér látható a 2. ábrán.



2. ábra - Predikciós tér és képek megoszlása a rácsokban (saját készítés)

## Training

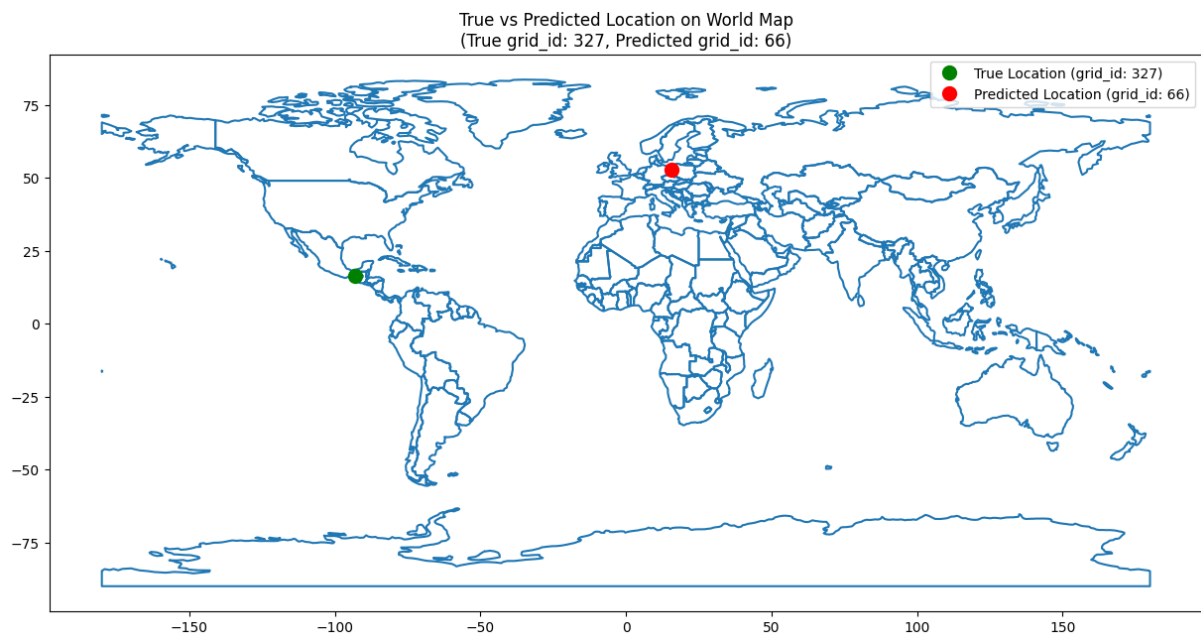
Mivel korábbi próbálkozások során felmerült a túlillesztés esélye (inkonzisztens elhelyezés, szűk predikciós tér), a teszt adatot a szokásosnál magasabb arányban, 30%-ban határoztam meg. A képek pixelértékeit pedig 0 és 1 közé normalizáltam, hogy a modell könnyebben tudja kezelni. A ResNet kezdeti súlyait az előre tanított imagenet adja meg, az eredeti klasszifikációs réteget eltávolítottam, hogy a saját képeimhez tudjam igazítani őket.

Az alapvető rétegek fagyasztásával csak a felső rétegeket tanítom, ami gyorsítja a tanulási folyamatot és megakadályozza, hogy az előre betanított súlyok megváltozzanak. Az „Adam” optimalizáló dinamikusan állítja be a tanulási rátát, ami gyorsabb és stabilabb konvergenciát

biztosít. A `categorical_crossentropy` veszteségfüggvény és az accuracy, Precision, Recall metrikák biztosítják a modell teljesítményének értékelését.

## Modell értékelés

Sajnos a modell ránézésre sem teljesít túl jól, gyakran tippel nagyon félre és továbbra is előforduló jelenség, hogy több képnél beakad egy adott rácsra.



3. ábra - Prediktált és valós pozíció (saját készítés)

Az training eredményei alapján a modell teljesítménye rendkívül alacsony. A pontosság (0.1057%) azt mutatja, hogy a modell előrejelzéseinek csupán egy töredéke volt helyes. A veszteség értéke 7.9617, ami magas hibaarányra utal.

## Korlátok

A projekt során elég sok nehezítő körülmény merült fel technikai oldalról:

- A modell tanítása sok időt vett igénybe, egy epoch lefutása másfél-két órát vett igénybe, így annak ellenére, hogy az epoch számának növelése javította a modell teljesítményét, kénytelen voltam lecsökkenteni 3 és 5 közötti értékekre.
- Bár a videokártyám elméletileg lehetővé teszi a párhuzamosítást, sok óra próbálkozás után sem sikerült működésre bírnom a CUDA-t, így CPU-val végeztem a tanítást, ami lényegesen lassabb volt.

- Bár kisebb mintán ellenőrizhető, hogy a modell értelmezhető eredményt ad-e, de a teljes adaton tanítás végére sokszor előfordult, hogy a tanítás végére a predikciók beragadtak egy pontra, vagy kontinensre, ezzel sok futási idő ment kárba.
- A hasonló, de jobban működő modelleket lényegesen több adaton tanítják, de a számítási kapacitás és tárhely hiányában erre nem volt lehetőségem.

Ezek alapján a projektet tovább lehetne fejleszteni az adatbázis növelésével (illetve a területi megoszlásuk optimalizálásával), a tanítás hatékonyabbá tételével, amit segíthetne a párhuzamosítás, vagy külön szerveren futtatás.

## Hivatkozások

- ayuseless. (2023). *Streetview Image Dataset* [Data set]. Kaggle.  
Letöltés helye: <https://www.kaggle.com/datasets/ayuseless/streetview-image-dataset>  
Letöltés ideje: 2024. 06. 16.
- Computer Vision Project. (2024). ResNet-50 – Deep Convolutional Neural Network.  
Letöltés helye: <https://computervisionproject.com/classification/resnet50.html>  
Letöltés ideje: 2024. 06. 16.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248-255). IEEE.  
Letöltés helye: <https://ieeexplore.ieee.org/document/5206848>  
Letöltés ideje: 2024. 06. 16.
- Mohan, S. (2023, April 8). *Keras implementation of ResNet-50 architecture from scratch*. Machine Learning Knowledge.  
Letöltés helye: <https://machinelearningknowledge.ai/keras-implementation-of-resnet-50-architecture-from-scratch/>  
Letöltés ideje: 2024. 06. 16.
- Zhang, L., Bian, Y., Jiang, P., & Zhang, F. (2023). A transfer residual neural network based on ResNet-50 for detection of steel surface defects. *Applied Sciences*, 13(9), 5260.  
Letöltés helye: <https://www.mdpi.com/2076-3417/13/9/5260>  
Letöltés ideje: 2024. 06. 16.