Interaktívna segmentácia obrazu pomocou Inside-Outside Guidance

Sabína Gregušová, Jan Šamánek, Adrián Tulušák
xgregu02, xsaman02, xtulus00

1 Úvod

V posledných rokoch sa začali rapídne rozvíjať práce zamerané na segmentáciu obrazu. Tento typ úlohy má potenciál nájsť uplatnenie v rôznych odvetviach, menovito v samoriadiacich vozidlách, analýze medicínskych či vzdušných snímkov, alebo v editácii videí či fotiek, a mnohých iných. Zaujímavou podúlohou pre segmentáciu obrazu je práve interaktívna segmentácia obrazu. Cieľom všeobecnej segmentácie je identifikovať a vysegmentovať všetky objekty v obraze na základe príslušnej triedy, zatiaľ čo interaktívna segmentácia sa zameriava na oddelenie jedného, užívateľom vybraného objektu *(foreground)*, od všetkého ostatného v obraze *(background)*.

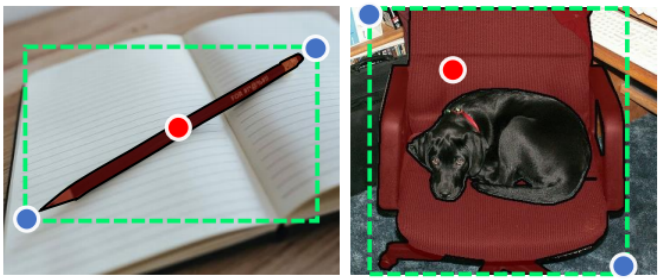
1.1 Existujúce riešenia

Jedným z prvých algoritmov, ktorý využíval deep-learning algoritmus bol *Deep Interactive Object Selection* pre interaktívnu segmentáciu obrazu predstavený v [6]. Tento článok jednoducho zhŕňa aj všetky predchádzajúce prístupy, napríklad dovtedy najznámejší [1], ktorý využíval algoritmus *Interactive Graph Cut*, na ktorý ďalej naviazali v článku [4] s optimalizovanou iteratívnou verziou. Tieto staršie algoritmy však veľmi záviseli na kvalite a hlavne množstve užívateľského vstupu, zatiaľ čo algoritmus *Deep Interactive Object Selection* veľmi zredukoval požadované množstvo užívateľského vstupu. Za užívateľský vstup prijímal *negatívne* (tam, kde sa objekt nachádza) a *pozitívneho* kliknutia (tam, kde sa objekt nenachádza), ktoré boli ďalej transformované do *Euklidovských distance máp*. Tento model bol natrénovaný pomocou FCN siete a dosahoval IoU až okolo 85% pri viac ako 6 kliknutiach.

Hoci tento algoritmus dosahoval dobré metriky, minimálne 6 kliknutí je pre bežného užívateľa stále dosť. Tento problém sa snaží riešiť algoritmus *Inside-Outside Guidance* v [7], ktorý môžeme považovať za jeden zo State-of-the art systémov, a na tomto prístupe je založená aj naša implementácia projektu.

2 Metóda

Náš tím si zvolil prístup *Inside-Outside Guidance* (ďalej iba IoG) [7], ktorého hlavným cieľom je zredukovať množstvo užívateľského vstupu a dosahovať metriky porovnateľné s predchádzajúcimi existujúcimi riešeniami. Podstatou IOG je najskôr získať *bounding box* pomocou dvoch kliknutí užívateľa (buď dvojica *horný ľavý roh, dolný pravý roh* alebo *horný pravý roh, dolný ľavý roh*) a zvyšné dve chýbajúce súradnice tohto obĺžnika je možné dopočítať. Užívateľ ďalej umiestni jedno kliknutie do vnútra objektu, ktorý chce vysegmentovať. Tento prístup ďalej umožňuje pridávať kliknutia aj po segmentácii obrazu a spresňovať ju, ak s ňou užívateľ nie je spokojný. Výhodou je, že bounding box nemusí úplne tesne obklopovať vybraný objekt. Kliknutia sú následne reprezentované ako v článku [3] pomocou *heatmapy*, kde bounding box predstavuje jednu heatmapu a zvyšné kliknutia druhú. Pri štandardnom RGB obrázku má teda výsledný obrázok 5 kanálov (3 pre RGB a 2 pre heatmapy).



Obr. 1: Ukážka použitia prístupu *Inside-Outside Guidance* pre užívateľský vstup. Prevzaté z [7].

Pri trénovaní siete nie je reálné, aby bol vstup získaný od skutočného užívateľa, ale musí byť náhodne vzorkovaný. Toto je implementované výberom bounding boxu z anotácie obrázku, ku ktorému je pripočítaný náhodný šum, aby vstup vyzeral ako od skutočného užívateľa. Kliknutie vo vnútri objektu je náhodne vygenerované vo vnútri objektu, s paddingom od okraja bounding boxu. Najväčšou výhodou tohto prístupu je schopnosť generalizovať aj iný typ predtým nevidených obrázkov bez potreby finetunovania.

3 Dataset

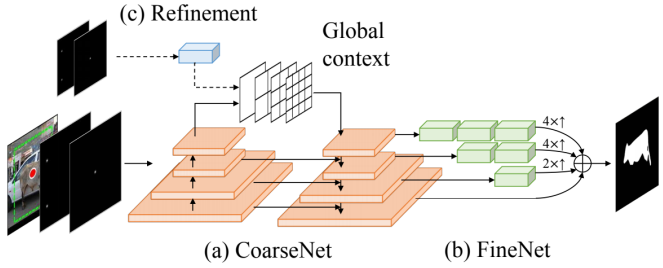
Momentálne existujú mnohé dostupné datasety určené pre segmentáciu obrazu, ktoré sa dajú adaptovať aj na interaktívnu segmentáciu. Medzi najčastešie používané patrí *Pascal*, *Grabcut*, *Berkley* alebo *MS Coco*; a sú najčastejšie používané pre validáciu a porovnanie presnosti medzi dnešnými state-of-the-art systémami.

Pre náš projekt sme sa rozhodli použiť dataset *MS Coco* pre evaluáciu aj trénovanie, pretože obsahuje cez 80 objektov vo viac ako 200 000 obrázkoch, ktoré zachytávajú bežné objekty zo života. Pre projekt bol použitý trénovací a validační dataset z roku 2017. Tento trénovací dataset obsahuje okolo 20 GB dát, ale pri trénovaní sa používala pravidelne obmieňaná polovica tohto datasetu kvôli obmedzeným výpočetným prostriedkom.

Obr. 2: Ukážka obrázkov z datasetu *MS Coco*. Prevzaté z [2].

4 Model

Model pre túto úlohu je založený na dvoch podsieťach: *CoarseNet* a *FineNet*. Takáto architektúra bola pôvodne navrhnutá pre odhadnutie pózy človeka v obrázku.



Obr. 3: Ukážka konceptu architektúry pre tento typ úlohy. Prevzaté z [7].

CoarseNet má v našom prípade upravenú architektúru veľmi podobnú *Resnet-50*, usporiadanej do pyramídovej štruktúry. Takáto sieť postupne aplikuje konvolúcie na vstupný batch obrázkov, na najhlbšej vrstve môže pridať *refinement heatmapu* pre globálny kontext. Následne na najhlbšej vrstve aplikuje dekonvolúciu, až kým sa nedostane na pôvodnú veľkosť batchu obrázku. Ako aktivačná funkcia pre všetky vrstvy je použitá ReLU.

FineNet je použitá na upsamplovanie medzivrstiev *CoarseNetu*, ktoré sú následne skonkatenované a tvoria výstupný obrázok. Jej cieľom je obnoviť chýbajúce detaily, ako napríklad hranice objektu. *FineNet* taktiež využíva ReLU na všetkých vrstvách okrem poslednej, kde je kvôli tresholdovaniu použitá sigmoida.

Pri trénovaní bol použitý learning rate s hodnotou 0.001, optimalizátorom *Adam* a chybová funkcia pre binárnu cross-entropiu.

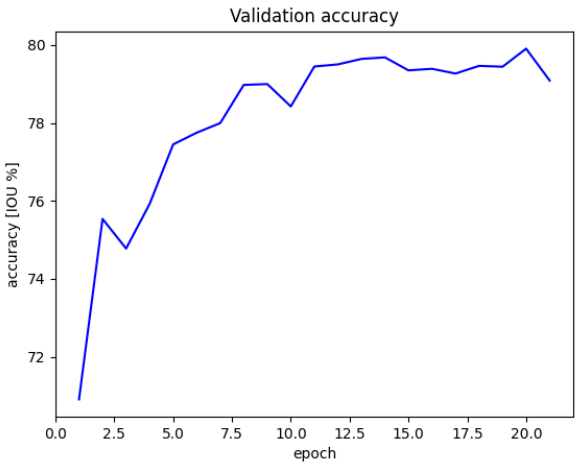
5 Evaluácia

Po každej epoche trénovania bol model evaluovaný na základe 3 metrík ([5]):

- *Pixel accuracy* - porovnáva, či sa jednotlivé pixely rovnajú; avšak táto metrika môže byť zavádzajúca, ak sú triedy na obrázku v nerovnováhe
- *Intersection over union (IoU)* - metrika, ktorá delí prienik očakávaného obrázku a predikcie modelu s ich zjednotením; najčastejšie používaná metrika pre porovnanie systémov
- *Dice coefficient* - metrika, ktorá používa dvojnásobok prieniku očakávaného obrázku a predikcie modelu podelený plochou oboch obrázkov v pixeloch

Experimentálne boli implementované všetky 3 metriky, avšak pre výber najlepšieho modelu bola použitá metrika IoU, keďže väčšina systémov používa práve túto metriku pre prezentáciu výsledkov.

Celkovo bol model natrénovaný pomocou 21 generácií. Výsledky boli pozitívne:



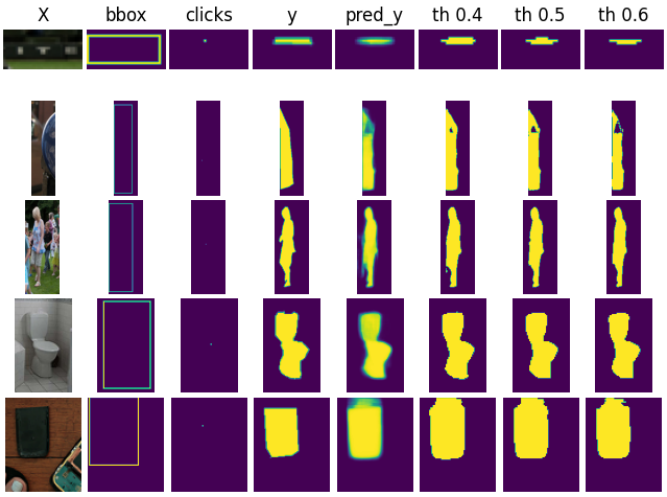
Obr. 4: Priebeh validačnej metriky IoU počas všetkých generácií trénovania

Ostatné metriky dosiahli taktiež dobrých výsledkov:

Metrika	Ohodnotenie
Pixel accuracy	92.307%
IoU	79.904%
Dice Coefficient	88.829%

Pôvodný článok pre IOG dosahoval približne 90% pre IoU, avšak je potrebné dodať, že využíval ako backbone *Resnet-101*, čo je sieť 2x tak veľká, ako bola použitá v tomto projekte, preto môžeme považovať IoU blížiace sa k 80% za veľmi slušný výsledok.

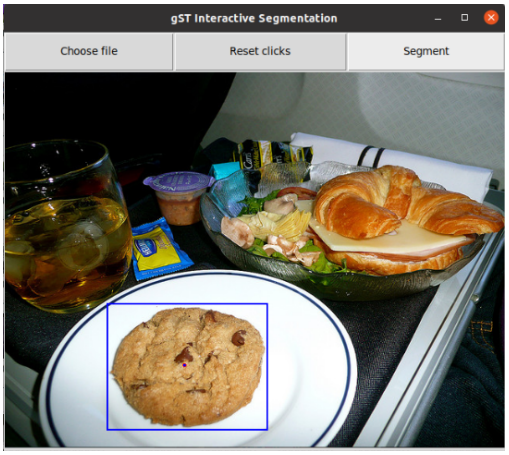
Nasledujúca ukážka prezentuje zobár vstupných obrázkov v RGB, vybraný bounding box, náhodne vygenerované kliknutie vo vnútri objektu, očakávaná segmentácia, predikcia modelu bez thresholdovania a predikcia modelu thresholdovaná na rôznych hodnotách. Vo finále je predikcia thresholdovaná na hodnote 0.5.



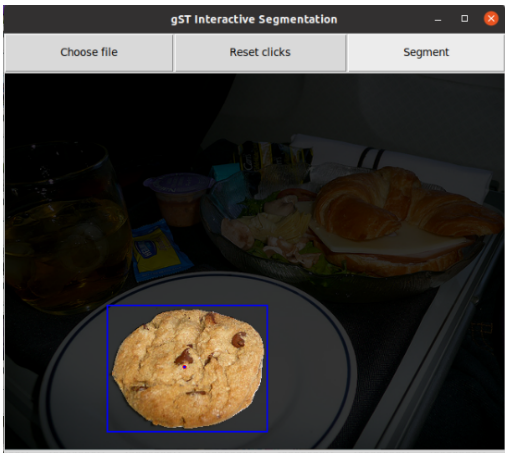
Obr. 5: Vizualizácia výsledkov segmentácie z nášho natrénovaného modelu.

6 Praktická aplikácia a GUI

Vrámci projektu bolo implementované jednoduché užívateľské rozhranie v jazyku `Python` pomocou modulu `tkinter`. Toto rozhranie umožňuje priamu interakciu skutočného užívateľa s obrázkom. Užívateľ si vyberie ľubovolný obrázok zo svojho pevného disku, pravým tlačítkom myši môže užívateľ umiestniť bounding box. Keďže sa môže na obrázku nachádzať iba jeden bounding box, po opätovnom kliknutí pravého tlačítka sa začne kresliť nový bounding box. Ľavým tlačítkom sa umiestňujú kliknutia vo vnútri objektu a užívateľ môže urobiť ľubovolne veľa kliknutí. Spravidla je segmentácia s väčším množstvom kliknutí presnejšia, najmä pri objektoch s nerovnomerným tvarom. Ak užívateľ urobil chybu v kliknutiach, môže ich zmazať tlačítkom `Reset Clicks`. Po zadaní užívateľského vstupu stačí stlačiť tlačidlo `Segment` a počkať na výsledok.



Obr. 6: Obrázok s vybraným užívateľským vstupom



Obr. 7: Výsledok segmentácie nášho najlepšieho modelu trénovaného 20 generácií

7 Ďalšie smery výskumu

Ak by boli pre trénovanie dostupné lepšie výpočetné zdroje, mohla by byť *CoarseNet* zmenená z *Resnet-50* na hlbšiu sieť *Resnet-101*, ktorá by síce vyžadovala ovela dlhšie trénovanie, ale pravdepodobne by dosahovala presnosť porovnateľnú so sieťou z pôvodného článku. Zároveň by v budúcnosti mohla byť validácia rozšírená aj na iné dostupné datasety, vrátane obrázkov, ktoré obsahujú predtým nevidené objekty.

Použitá literatúra

[1] Boykov, Y.; Jolly, M.-P.: Interactive graph cuts for optimal boundary and; region segmentation of objects in N-D images. *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, doi:10.1109/iccv.2001.937505.

[2] Lin, T.-Y.; Maire, M.; Belongie, S.; aj.: Microsoft COCO: Common Objects in Context. *Computer Vision – ECCV 2014*, 2014: str. 740–755, doi:10.1007/978-3-319-10602-1_48.

[3] Maninis, K.-K.; Caelles, S.; Pont-Tuset, J.; aj.: Deep Extreme Cut: From Extreme Points to Object Segmentation. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, doi:10.1109/cvpr.2018.00071.

[4] Rother, C.; Kolmogorov, V.; Blake, A.: "GrabCut". *ACM SIGGRAPH 2004 Papers on - SIGGRAPH '04*, 2004, doi:10.1145/1186562.1015720.

[5] Tiu, E.: Metrics to Evaluate your Semantic Segmentation Model. Oct 2020.
URL `https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2`

[6] Xu, N.; Price, B.; Cohen, S.; aj.: Deep Interactive Object Selection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, doi:10.1109/cvpr.2016.47.

[7] Zhang, S.; Liew, J. H.; Wei, Y.; aj.: Interactive Object Segmentation With Inside-Outside Guidance. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, doi:10.1109/cvpr42600.2020.01225.