

Interaktivní segmentace RGB-D obrázků

Sobol Jan <xsobol04@stud.fit.vutbr.cz>
Endrych David <xendry02@stud.fit.vutbr.cz>
Dovičic Denis <xdovic01@stud.fit.vutbr.cz>

29. prosince 2021

1 Úvod

Segmentace obrazu v počítačovém vidění je důležitá technika používaná k rozpoznání objektů, což přispěje k lepšímu pochopení scény. Jednat se může o segmentaci objektu oproti zbytku scény nezávisle na jeho druhu. V tomto případě jde o binární klasifikaci do dvou tříd: popředí a pozadí. Jinou možností je sémantická segmentace, u které je úkolem klasifikátoru určit kromě hranice objektu také jeho druh – třídu. Přičemž objektů různých tříd se v obraze naráz vyskytuje několik.

Strojová automatická segmentace způsobuje chyby v závislosti na kvalitě klasifikátoru. V některých odvětvích, ve kterých je žádoucí využít metody počítačového vidění, ale není možné odchylky tolerovat, případně se alespoň musí provést manuální korekce. Pro tyto případy existují metody předpokládající interaktivní zásah uživatele. Interaktivní segmentace se dále dělí na manuální a poloautomatizovanou, kde uživatel koriguje automatickou segmentaci. Z dnešního pohledu jsou atraktivní právě poloautomatizované metody, které najdou využití při zpracování medicínských dat nebo při anotaci fotografií a videa.

Jeden z nejznámějších algoritmů pro poloautomatizovanou segmentaci se nazývá graph-cut. Algoritmus je postaven na minimalizaci max-flow/min-cut energie při řezu grafem. V současnosti se problém přesunul na pole hlubokých konvolučních neuronových sítí, jejichž výhoda spočívá ve schopnosti pochopit sémantiku scény a zpracování grafu již bývá využito pouze pro optimalizaci výstupního obrazu.

V této práci je prezentována metoda poloautomatizované interaktivní segmentace využívající hluboké neuronové sítě. Navazuje na způsob trénování, předzpracování uživatelských interakcí a inspiruje se typem architektury sítě popsané v [6] a [5]. Na rozdíl od těchto metod však neuronová síť využívá informace o hloubce, přesněji disparitní mapy spárované s RGB zdrojovými obrazy.

2 Teorie

Uživatelské interakce při klasifikaci na popředí/pozadí se dělí na dva druhy: pozitivní a negativní. Pozitivní jsou uživatelem umisťovány na segmentovaný objekt, negativní na jeho okolí. Předpokládá se postup: uživatel umístí první pozitivní interakci na segmentovaný objekt. Tím zahájí segmentaci. Dále umisťuje negativní značky na místa, která označila neuronová síť za objekt, ačkoliv ním není (tzv. *false positives*). Pokud je to potřeba, přidává uživatel pozitivní značky na místa, kde se odchylila segmentovaná hranice a hranice objektu (tzv. *false negatives*).

Tento proces musí být simulován při trénování konvoluční neuronové sítě. Způsob výběru nové interakce může být náhodný nebo řízený pravděpodobností, která vychází z běžného chování uživatele. Uživatelské interakce typicky nejsou sítí předávány jako jeden obrazový bod, ale jako pravděpodobnostní mapa, kde má místo interakce nejvyšší pravděpodobnost. Pravděpodobnost se snižuje se vzdáleností od bodu uživatelské interakce.

3 Příprava datové sady

V rámci této práce byla použitá datová sada Cityscapes [1], která nabízí RGB obrázky, hloubkovou mapu a target (viz. obrázek 1). Pro účely projektu bylo nutné datovou sadu transformovat do formátu, ve kterém je možné síť učit segmentovat RGB-D obrázky podle uživatelských interakcí. To je segmentovat jen objekt vybraný uživatelem, případně umožnit opravu segmentace pokud objekt nebyl korektně segmentovaný.

Na dosáhnutí tohoto cíle bylo nutné target obrázky s popisem scén rozdělit do více obrázků, které obsahující vždy právě jeden objekt (viz. obrázek 2). Třídy datové sady byly také omezené pouze na třídy označené hodnotami 24 až 33, protože ostatní nemají k dispozici označení konkrétního objektu.



(a) RGB obrázek

(b) Hloubková mapa

(c) Target

Obrázek 1

ních instancí. Pokud by se nacházelo více objektů v těsné blízkosti, tak by je poté nebylo možné rozpoznat, která instance je platná [1]. Další manipulace s datovou sadou spočívá ve změně rozlišení r -krát, z důvodu příliš vysokého rozlišení původních obrázků 1024×2048 pixelů.



(a) Target s popisem
scény.

(b) Target pouze s jedním
objektem

(c) Target pouze s jedním
objektem

Obrázek 2: Ukázka transformace targetu (a) do targetů potřebných v tomto projektu, (b) a (c)

K učení neuronové sítě interaktivně segmentovat RGB-D obrázky je potřebné kromě RGB obrázku, hloubkové mapy a targetu přidat ještě dva kanály představující pozitivní a negativní uživatelské interakce. Pozitivní interakce představují volbu segmentovaného objektu uživatelem a negativní interakce umožňují uživateli opravit nekorektní segmentace. Při generování těchto interakcí, na základě [5], předpokládáme zkušeného uživatele, který kliká vždy do středu objektů, případně do středu špatně segmentovaných oblastí. Interakce je možné generovat následujícím způsobem:

$$D = G - P = \begin{cases} +1 & G = 1 \text{ a } P = 0, \\ -1 & G = 0 \text{ a } P = 1, \\ 0 & jinak, \end{cases} \quad (1)$$

kde G je target a P je predikce sítě. Po získání mapy D je možné získat regiony segmentovaných oblastí neodpovídajících objektu F_{pos} a regiony chybějících

segmentací, kde se objekt ve skutečnosti nachází F_{neg} :

$$F_{pos}(x, y) = \begin{cases} 1 & D(x, y) > 0, \\ 0 & jinak \end{cases} \quad (2)$$

a

$$F_{neg}(x, y) = \begin{cases} 1 & D(x, y) < 0, \\ 0 & jinak. \end{cases} \quad (3)$$

Tyto mapy špatně segmentovaných oblastí F_{pos} a F_{neg} , je následně nutné transformovat pomocí vzdálenosti (např. pomocí chamfer distance, kde jako features jedné skupiny se zvolí souřadnice, kde $F_{pos/neg}(x, y) = 1$ a features druhé $F_{pos/neg}(x, y) = 0$) díky čemuž získáme nejstřednější pixel M_{pos} a M_{neg} segmentovaných oblastí. Tento pixel je považovaný za uživatelskou interakci (viz. obrázek 3). [5]



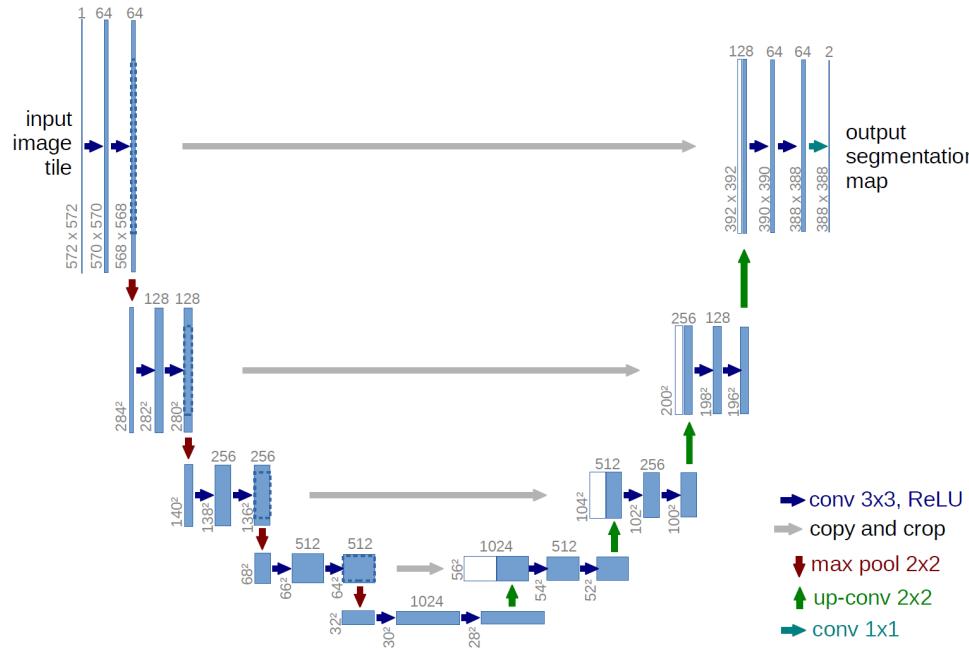
Obrázek 3

Posledním krokem je vytvořit nové nulové mapy I_{fpos} a I_{fneg} velikosti segmentovaného obrázku s hodnotou jedna nastavenou na pozici nejstřednějšího pixelu M_{fpos}/M_{fneg} a aplikovat na nich gaussian blur s určitým parametrem $sigma$. [5]

4 Architektura neuronové sítě

Pro naši síť jsme si zvolili architekturu zvanou U-Net [4], kterou lze vidět na obrázku 4. Jedná se o jednu ze základních architektur plně konvolučních neuronových sítí, které jsou určeny pro segmentaci obrazu. U-Net si našel uplatnění při zpracování obrazu v biomedicíně např. pro segmentaci snímku mozku [2]. Tahle architektura se skládá ze dvou částí. V první části, která se nazývá contracting path dochází k postupnému zmenšování vstupních rozměrů. Díky tomu je síť schopna získávat informace o kontextu z větší oblasti.

Druhá část sítě se nazývá expansive path a ta upřesňuje lokalizaci. V této části dochází k opětovnému zvětšování rozlišení a k připojení s uloženými vrstvami z contracting path. V poslední vrstvě dochází k mapování kanálů na počet odpovídající počtu klasifikačních tříd. Jako chybovou funkci jsme zvolili Dice Loss [3], která je založena na koeficientu Dice 5. Ten udává míru překrývání dvou vzorků. V tomhle případě zkoumáme míru překrývání výstupu sítě a targetu, který je vygenerovaný z ground truth datové sady. Jelikož koeficient nabývá při nejlepších výsledcích hodnot blízké hodnotě jedna, tak výpočet hodnoty chyby je určena jako 4.



Obrázek 4: Architektura U-Net [4]

$$DICE\ LOSS = 1 - DICE \quad (4)$$

$$DICE = \frac{2|A \cap B|}{|A| + |B|} \quad (5)$$

5 Trénování

Jelikož je datová sada rozdělená do německých měst, trénovaní neuronové sítě z kapitoly 4 probíhá po b_{size} náhodně zvolených obrázcích z jednotlivých měst. Po zvolení b_{size} náhodných RGB-D obrázků probíhá r -krát zmenšení. Následně je k tomuto batchi generovaných I_{max} interakcí I_{fneg} a I_{fpos} postupem popisovaným v kapitole 3. Výsledné data mají složení $\{r, g, b, d, I_{fpos}, I_{fneg}\}$, kde r , g a b jsou kanály RGB obrázku a d je hloubková mapa. Jednotlivé interakce jsou generované real-time odevzou modelu následným postupem:

1. První interakce I_{fneg} se nachází ve středu aktuálního objektu. I_{fpos} obsahuje samé nuly.
2. Data $\{r, g, b, d, I_{fpos}, I_{fneg}\}$ předej síti a získej predikce P .
3. Pokud počet interakcí n dosáhl I_{max} , pokračuj bodem 8.
4. Spočítej mapy F_{neg} a F_{pos} . Pro mapu s větším počtem označených pixelů vygeneruj novou interakci I_s , kde $s = pos$ anebo $s = neg$.
5. Vynásob I_s váhovým faktorem w , kde $w = (I_{max} - n + 1) / I_{max}$.
6. Připočítej I_s k mapě interakcí z předešlé iterace I_s .
7. Pokračuj bodem 2.
8. Spočítej Dice loss z kapitoly 4 a skonči.

Všechny data jsou před předáním síti normalizované do rozsahu $(0, 1)$.

6 Grafické uživatelské rozhraní

Pro testování natrénované neuronové sítě byla vytvořena jednoduchá GUI aplikace pomocí knihovny OpenCV. Aplikace umožňuje načíst RGB obrázek a disparitní mapu. Soubor s obrázkem a disparitní mapou musí mít stejný název s koncovkou `_leftImg8bit.png`, resp. `_disparity.png`. To vychází z pojmenování souborů v datasetu Cityscapes (viz kapitola 3). 2D data jsou následně zmenšena na rozlišení, které požaduje na vstupu neuronová síť.

GUI se ovládá stisky kláves a myši. Kliknutí do obrázku levým tlačítkem myši vytvoří pozitivní interakci, pravé tlačítko umístí interakci negativní.

Stisk levého tlačítka musí pro zahájení segmentace proběhnout jako první. Výstupní maska neuronové sítě je následně přepracována po každém stisku tlačítka myši a vizualizována změnou odstínu barev (popředí se zbarví do červena, pozadí do modra). Pro vymazání všech uživatelských interakcí a reset segmentace se stiskne klávesa R. Ukončení aplikace proběhne po stisku klávesy Esc.

```
working dir: gui/
usage: app.py [-h] -i IMG [-m MODEL] [-d]

-h, --help            show this help message and exit
-i IMG, --img IMG    image name (without _leftImg8bit.png)
-m MODEL, --model MODEL stored model parameters
-d, --debug          debug mode
```

Ve zdrojovém kódu je samotná metoda segmentace oddělena ve třídě **UNet Segmentation**. Instance této třídy je předána třídě ovládající okno aplikace. GUI aplikaci tak lze rozšiřovat o jiné techniky nebo modely neuronových sítí. Například U-Net segmentace s optimalizací graph cut a bez optimalizace.



Obrázek 5: Demonstrační GUI aplikace

7 Vyhodnocení

Parametr	Hodnota	Poznámka
b_{size}	8	velikost batche
r	4	zmenšení
$sigma$	2	gaussův šum
I_{max}	6	maximální počet interakcí

Tabulka 1: Parametry trénování a transformace datové sady.

Pro vyhodnocení využíváme metriku Mean Intersection over Union a pixel accuracy. První z metrik je určující pro validaci modelu. Vyhodnocení nebylo z časových důvodů dokončeno. Výsledky budou po dokončení vyhodnocení aktualizovány.

8 Závěr

Z důvodu časové tísň a neočekávané náročnosti úprav provedených na datasetu se nepodařilo natrénovat neuronovou síť na úroveň, která by dostačovala k ověření segmentace v GUI. V rámci trénování bylo zjištěno, že síť nejprve začne segmentovat často se vyskytující objekty (automobil). V této fázi nerozumí instancím – v obraze klasifikuje jako popředí všechny výskytu objektu. Neuronová síť nejprve uživatelské interakce úplně ignoruje. Po dlouhé době tréninku (vice než 2h) začne predikci podle interakcí nepatrнě upravovat.

Reference

- [1] CORDTS, M., OMRAN, M., RAMOS, S., REHFELD, T., ENZWEILER, M., BENENSON, R., FRANKE, U., ROTH, S., AND SCHIELE, B. The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), IEEE, pp. 3213–3223.
- [2] DONG, H., YANG, G., LIU, F., MO, Y., AND GUO, Y. Automatic brain tumor detection and segmentation using u-net based fully convolutional networks. *CoRR abs/1705.03820* (2017).
- [3] MILLETARI, F., NAVAB, N., AND AHMADI, S. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *CoRR abs/1606.04797* (2016).
- [4] RONNEBERGER, O., FISCHER, P., AND BROX, T. U-net: Convolutional networks for biomedical image segmentation. *CoRR abs/1505.04597* (2015).
- [5] SAKINIS, T., MILLETARI, F., ROTH, H., KORFIATIS, P., KOSTANDY, P. M., PHILBRICK, K., AKKUS, Z., XU, Z., XU, D., AND ERICKSON, B. J. Interactive segmentation of medical images through fully convolutional neural networks. *CoRR abs/1903.08205* (2019).
- [6] XU, N., PRICE, B. L., COHEN, S., YANG, J., AND HUANG, T. S. Deep interactive object selection. *CoRR abs/1603.04042* (2016).