

Ročníkový Projekt

Specifikace

Václav Mlejnský

1 Téma

Cílem mého ročníkového projektu je vytvořit neuronovou síť, která jako vstup dostane počáteční a koncovou část videa a vygeneruje chybějící prostřední část. Tato vygenerovaná část bude navazovat na konec počáteční části a na začátek koncové části. Síť tedy bude schopna dogenerovat prostředek videa. Celý projekt je rozdělen na více částí. Každá část staví na předchozích.

2 První část

2.1 Cíl

Rozpoznání lineárního pohybu cílového pixelu ze dvou navazujících framů a dogenerování třetího framu, který přímo navazuje na druhý frame.

2.2 Vstup

Vstupem jsou černobílá videa. Video obsahuje právě jeden cílový pixel a bílé pozadí. Cílový pixel je černý a jeho pohyb probíhá v jednom řádku či sloupci. Tento pixel se hýbe lineárně od jednoho okraje k protějšimu - zleva doprava, shora dolů, zprava doleva nebo zdola nahoru. Pro dané video s rozlišením $m * n$ můžeme mít až $2 * (m + n)$ různých videí výše popsaného formátu.

2.3 Zpracování vstupu

Síť na vstupu nedostane pouze tři framy. Vstup sítě tvoří vždy až celá batch, to je několik zpracovaných videí zároveň. Jedno video je na vstupu zpracováváno sekvenčně. Vždy vezmeme tři, na sebe navazující framy. První dva framy se vloží do pole, které reprezentuje vstup sítě (to jsou hodnoty x -ové sítě). Třetí frame se vloží do pole reprezentující očekávaný výsledek sítě pro předchozí dva framy (to jsou hodnoty y -ové sítě).

2.4 Modifikace I - Změna pozadí

Zesložiti vstupních videí tím, že změníme jejich pozadí. Čistě bílé pozadí nahradíme černobílým šumem.

2.5 Modifikace II - Přejchod z černobílého modelu na RGB model

Pozadí i cílový pixel mohou mít různé barvy. Dataset tvoří videa obsahující kombinace barev cílového pixelu a pozadí.

2.6 Modifikace III - Barevný šum

Jde o kombinaci Modifikace I a Modifikace II. V barevném videu z Modifikace II nahradíme jednobarevné pozadí barevným šumem.

3 Druhá část

Model sítě staví na první části.

3.1 Cíl

Rozpoznání nelineárního pohybu cílového pixelu ze dvou navazujících framů a dogenerování třetího framu, který přímo navazuje na druhý frame.

3.2 Vstup

Forma vstupu zůstává stejná jako v první části. Mění se pohyb cílového pixelu. Nyní se nejedná o lineární pohyb, ale o pohyb nelineární. Cílový pixel se může v jednom videu hýbat ve více směrech. Jde o takový pohyb, že cílový pixel nepřeskočí více pixelů naráz v jednom framu. V každém framu si může cílový pixel vybrat jeden ze čtyř směrů (doleva, nahoru, doprava, dolů), ve kterém se posune právě o jeden pixel v dalším framu.

3.3 Zpracování vstupu

Princip zpracování vstupu zůstává stejný jako v první části.

3.4 Modifikace I, II, a III

Myšlenka těchto modifikací je stejná jako v první části. Tentokrát se ale jedná o kombinaci s nelineárním pohybem cílového pixelu.

4 Třetí část

Model sítě staví na předchozích dvou částech

4.1 Cíl

Dogenerovat prostřední část videa na základě počáteční části a koncové části videa.

4.2 Vstup

Vstupem je video, které má stejný základ jako v druhé části - černobílé video a cílový pixel provádějící nelineární pohyb.

4.3 Zpracování vstupu

Jednotlivé framy videa zpracováváme sekvenčně - video procházíme frame po frame. Každých n framů si uložíme do pole a pak toto pole rozdělíme na tři části - počáteční, prostřední a koncovou. Počáteční a koncová část jsou stejně dlouhé, prostřední část může být kratší. Počáteční a koncová část jsou vstupy sítě (to jsou hodnoty x sítě). Prostřední část je očekávaný výstup sítě pro danou počáteční a koncovou část videa (to jsou hodnoty y sítě).

4.4 Modifikace I, II a III

Myšlenka těchto modifikací zůstává stejná jako v předchozích částech. S tím rozdílem, že vstup zpracováváme, jak je popsáno výše.

5 Dodatek ke všem částem

5.1 Vstup

Vstup si v každé části generuji sám.

Rozlišení videí, jejich délka a FPS jsou proměnné, které se ustálí v průběhu tvorby sítě. Záleží totiž, jak bude zpracovávání videa s daným rozlišením výpočtově náročné. Od rozlišení videa se také odvíjí počet různých videí v datasetu.

Velikosti jednotlivých částí videa ve třetí části jsou proměnné, které se ustálí v průběhu řešení problému.

5.2 Model sítě

Základem každého modelu bude nejspíše alespoň jedna konvoluční vrstva. Konvoluční vrstvy jsou specificky vhodné pro zpracování obrazu.

5.3 Dataset

Dataset bude rozdělen na tři části - trénovací část, krosvalidační část a testovací část. Trénovací část bude tvořit 60% celého datasetu, krosvalidační 20% a

testovací část zbylých 20%.

Krosvalidační část slouží pro správné určení hyperparametrů neuronové sítě. Dále je krosvalidační část užitečná k určení, jestli nedošlo k přetrénování sítě na trénovacích datech či k nedostatečnému natrénování. To by vedlo ke špatné generalizaci sítě.

6 Technické řešení

- Programovací jazyk, který použiji pro tvorbu videí, jejich zpracování a tvorbu neuronové sítě je jazyk Python
- Pro ukládání dat do polí a manipulaci s nimi použiji Python knihovnu [NumPy](#)
- Pro práci s videem a obrazem využiji knihovnu [OpenCV](#)
- Pro vytvoření sítě použiji knihovnu [Tensorflow](#)