



Kognitivní systémy – Semestrální práce

*Vytvořte program, který z kamery pozná, jaká čísla
padla na dvou hozených kostkách*

Jiří Němeček, Václav Smítka
nemecj38@fel.cvut.cz, smitkva1@fel.cvut.cz

FAKULTA ELEKTROTECHNICKÁ,
OTEVŘENÁ INFORMATIKA

11. ledna 2022

Obsah

1	Zadání	1
2	Popis problému	2
2.1	Příprava prostředí	2
2.2	Fotografie prostředí	3
2.3	Analýza fotografie	3
2.4	Neuronová síť	4
3	Podobnosti a odlišnosti vůči lidskému mozku	6
4	Závěr	7

1 Zadání

Vytvořte program/modul pro robota, který z kamery pozná, jaká čísla padla na dvou hozených kostkách

Úloha zaměřená na vidění a pozornost. Využijte buď vlastní nebo libovolný online dataset se dvěma tečkovanými kostkami a vytvořte modul, který spočítá celkový počet teček na svrchních stranách kostek, a to i na nových obrázcích, které jsou použité jen pro evaluaci. Která mozková centra se u člověka při takové úloze zapojují? Je možné se v nich nějak inspirovat?

2 Popis problému

Jak zadání naznačuje jedná se o problém, blízký počítačovému vidění. Je úkolem programu detekovat na obrázku dvě házeací kostky a spočítat kolik bylo hozeno na obou dohromady. Úkol na první pohled působil možná až triviálně, ale v konečném důsledku jsme si uvědomili, že to má své zákonitosti.

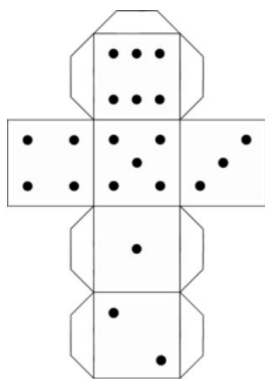
Úkol samotný lze rozdělit do několika po sobě jdoucích kroků. Prvním úkolem je potřeba vytvořit prostředí v kterém bude možné postup provést. Zároveň je třeba vybrat vhodné házeací kostky. Po hodu kostkami je třeba zhotovit fotografii nebo natočit daný prostor obsahující obě kostky. V poslední řadě je třeba analyzovat daný obraz a vyhodnotit ho.

2.1 Příprava prostředí

Na základě našich zkušeností z průběhu této práce jsme zjistili několik základních kritérií, které musí prostředí splňovat pro úspěšné vykonání součtu hozených čísel. Tyto kritéria vznikla na základě obtížnosti detekce kostky na obrázku.

V ideálním případě by v prostoru neměla být ostrá světla, aby kostky neházely ostré stíny. Zároveň by povrch neměl být příliš lesklý nebo by fotografie neměla obsahovat zrcadla, která by odrazela kostky. Odrazy a odlesky mohou vykazovat podobné vizuální vlastnosti jako samotné kostky, a to by vedlo k chybným součtům.

Velmi důležité je i zvolení správných barev. Barva povrchu, kostky a teček kostky by měly být od sebe rozdílné a ideálně dostatečně kontrastní. Zároveň všechny tečky na kostkách by měli být nepoškozené a vizuálně odpovídat klasickým schémátům. Pracovali jsme s kostkami, které splňují pravidlo, že součet protilehlých stran je právě 7. Kostka by též měla být čistá a ideálně jednobarevná po svém celém povrchu.



Obrázek 1: Plášť hrací kostky

Hod kostkou by měl být vykonán tak, aby kostky zůstali ležet nějaký kus od sebe. Zvýší to pravděpodobnost na správnou detekci kostek.

2.2 Fotografie prostředí

Uložení konkrétní konfigurace v podobě fotografie není obtížný úkol. Je možné využít jak fotoaparátu nebo video kamery. Zároveň tato část je klíčová pro závěrečnou část úkolu. Čím kvalitnější fotografie bude tím bude snažší poslední krok. Nezaostřený obrázek nebo máznutí ovlivňuje detekci teček kostek, a to vede k chybnému závěru.

Hraje roli také úhel, z kterého je fotka pořizována. Jednak to ovlivňuje počet stran kostky, které jsou na fotografii vidět, ale také pod určitým úhlem mohou být tečky v zákrytu, a to vede k chybnému závěru. Například konfigurace teček pro číslo 6 je náchylná na nadměrný úhel z obou stran. Zároveň lze pořídit fotografii tak, aby nebyla vidět vrchní strana kostky, což by znemožnilo součet úplně. Na druhou stranu, když by bylo možné vykonávat fotografie ze shora – kolmo na podložku, tak to výrazně zjednoduší výslednou analýzu fotografie.



Obrázek 2: Číslo šest pod velkým úhlem

Vzdálenost objektivu od kostek hraje roli. Ve chvíli, kdy má fotografie nízké rozlišení, výsledná analýza teček je za těchto okolností prakticky nemožná.

2.3 Analýza fotografie

Došli jsme k závěru, že je možné analyzovat fotografie dvěma způsoby. První způsob, který jsme zkusili byl čistá analýza fotografie. Použili jsme k tomu OpenCV knihovnu pro programovací jazyk Python. Druhým způsobem bylo na základě datasetu fotografií s kostkami natrénovat systém, který bude schopný detekovat kostky a určit jaké číslo bylo hozeno.

První způsob, přinesl zjištění, že se jedná o poznání složitější cestu než v případě naučení umělé inteligence. Problém je možné rozdělit do několika pod částí, která sebou nesla každá své problémy. První fáze spočívá v detekci kostky v prostoru. Na to existují již nejrůznější algoritmy postavené na bázi detekci hran v obrazu nebo na detekci barevných ploch. Zde byl velký problém, pokud kostka nebyla dostatečně kontrastně odlišena od okolí. Druhá fáze spočívá v detekci jednotlivých teček kostky. Zde též jsme se dostali do úzkých a komplikací v mnoha ohledech.

Na detekci teček na kostce je možné použít hned několik algoritmů, které mají své specifika. První možností je detekce hran, kde vzniká problém s možným nízkým rozlišením a

případným správným nastavením thresholdů, které často závisí na konkrétní fotografii. To samé v případě snahy detekce teček podle barev. Barvy mezi sebou často obsahují na fotografii nějaký přechod, fotografie bývají zašumělé a to výrazně zhoršuje schopnost detekce správných teček.

Naším cílem bylo detekovat elipsy, protože to jsou tvary, které fyzicky odpovídají tečkám na fotografii v případě jakéhokoliv natočení kostky nebo objektivu. OpenCV má v sobě zabudované dvě funkce, které umožní buď detekci elips nebo vložení elipsy do konkrétních třech bodů. Druhá možnost je též značně komplikovaná k využití, protože hraje velkou roli, jakou konkrétní body by byly zvoleny.

Na základě pokusů jsme došli k závěru, že toto řešení není ve své jednoduchosti možné a bylo by potřeba zvolit nějaký sofistikovanější způsob detekce elipsy nejspíše s podmínkou určitého rozlišení (počtu pixelů kostky) podle volby techniky. Nejspíš by bylo potřeba zvolit vlastní detekci sousedních pixelů a celkově pracovat na mnohem nižší úrovni s konkrétní fotografií.

Dalším úkolem, po korektní detekci teček je detekce stěn kostky a určení vrchní strany. To je v případě elips možné na základě natočení os. Zároveň tento úkol nemusí být přímočarý z důvodu chyb detekce jednotlivých elips. Víme, že v základu každá tečka je kolečko a delší poloosa značí ubíhající stěnu. Elipsy s podobným natočením značí tečku na stejné straně. To bohužel nefunguje v každém případě. Nemáme naměřené statistiky úspěchu, ale s pokusy, co jsme zkoušeli jsme nebyli spokojeni.

Velká míra zkreslení na základě nedostatků fotografie nebo kostek vedla k ukončení pokusů s tímto způsobem řešení a zaměřili jsme se na způsob druhý. Zde bylo klíčové vytvořit dataset s kostkami ze všech možných stran a nechat systém naučit na tyto kostky.

2.4 Neuronová síť

Druhý, úspěšnější přístup byl skrze použití neuronových sítí. Použili jsme neuronovou síť YOLOv5, časem ozkoušený model pro detekci objektů. Pro jeho použití jsme potřebovali dataset. Ten jsme vytvořili pomocí nástroje Roboflow, na obrázcích z datasetu, který byl doporučen v zadání, některých dalších fotek z internetu a také našich vlastních fotek, celkem 442 obrázků s 2010 kostkami.

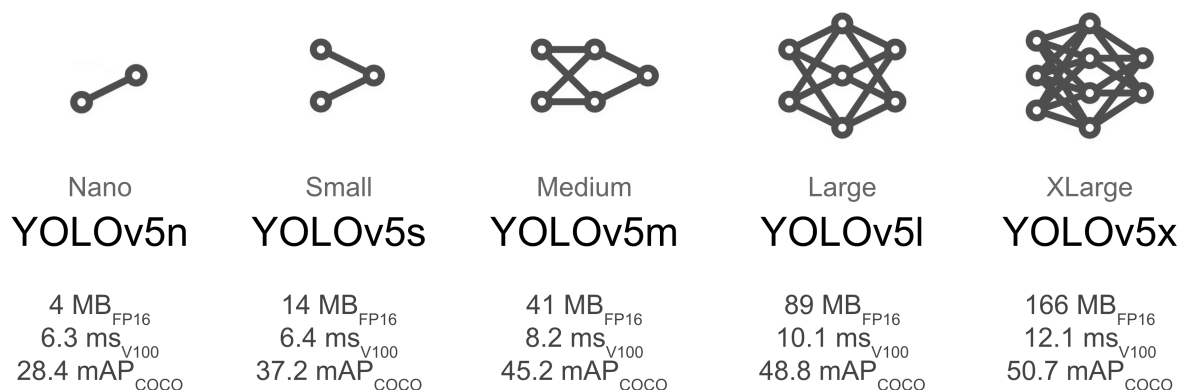
Class Balance



Obrázek 3: Zastoupení tříd v obrázcích

Tyto data jsme rozdělili na trénovací, validační a testovací data, v poměru 70:20:10. Trénovací data byla nadále augmentována pomocí přidání šumu, změny barev, ořezání, a rozmazání. Celkem se trénovalo na více než 800 obrázcích.

Natrénovali jsme každou z 5 velikostí. Jelikož nemám na svém osobním počítači přístup k GPU, časy inference se hodně liší od těch proklamovaných autory. Proto se čas ukázal jako důležitý faktor při výběru nejlepšího modelu.



Obrázek 4: Velikosti YOLOv5 modelů

Vybrali jsme model střední velikosti, učil se na GPU v cloudu 5 a půl hodiny, dosáhl obstojných výsledků, druhých nejlepších v celkovém měřítku.

V reálné aplikaci zvládá téměř 2 snímky za sekundu na mém osobním počítači. To není mnoho, nicméně pro zadaný úkol to stačí, pokud jde o házení kostkami před kamerou. Demo je možné shlédnout v repozitáři (github.com/Epanemu/GameDiceRecognition).

	1	2	3	4	5	6	dohromady	rychlost
nano	0.86	0.67	0.84	0.84	0.93	0.84	0.853901	100 ms
small	0.86	0.78	0.88	0.95	0.95	0.99	0.941099	260 ms
medium	0.79	0.78	1.00	0.89	0.95	0.99	0.945934	600 ms
large	0.79	0.78	0.88	0.95	0.95	0.99	0.935714	1080 ms
x-large	0.93	0.89	0.92	0.84	0.98	0.99	0.952857	1900 ms

Tabulka 1: Srovnání přesnosti a rychlosti různě velkých modelů na testovací sadě

3 Podobnosti a odlišnosti vůči lidskému mozku

Člověk zpracovává a vyhodnocuje informace v mozku. Tento orgán obsahuje dvě hemisféry, které se obě podílejí na tomu, čemu my říkáme pozornost. Naše úloha vyžaduje interpretaci prostředí a detekci předmětů. Člověk má dva systémy pozornosti a v zajímá nás především posterionální systém, který používáme pro vizuoprostorové úlohy.

Další vlastnost mozku je selektivní zaměření senzorického aparátu. To můžeme v našem případě využít též. Náš program totiž nezajímá nic víc než předměty, které vypadají jako kostky. Zbytek světa můžeme opomenout. Stejně jako mozek, tak i naše neuronová síť má jistou představu o tom, co je to kostka, a to též využíváme k detekci kostek v prostoru.

Oproti našemu systému mozek nepotřebuje se naučit všechny možné polohy kostek a jejich variace. Stačí rozpoznat na povrchu tečky a ty poté jednoduše sečte. Pokud je to mozek s kostkami obeznámený, tak pak již nepočítá tečky, ale jednoduše na základě charakteristických seskupení teček hned ví, o které číslo se jedná. To nám pro náš program přišlo značně složitější na přípravu a implementaci.

Důležitá "nevýhoda" mozku je, že se musí zaměřit na každou kostku zvlášť, zatímco kamera má zaostřené všechny najednou. Počítač také rychleji sčítá, celkově se tedy výsledku dobere rychleji, zvláště pro více kostek.

4 Závěr

Úkolem bylo vytvořit program, který dokáže rozpoznat dvě kostky v prostoru a zároveň říct, kolik na nich padlo dohromady. Účelem bylo zvládnout vyzkoušet si jakým způsobem počítače/roboti vidí prostor pomocí objektivu a jakým způsobem lze pracovat s informacemi, které obraz obsahuje.

Zjistili jsme, že problém, který je pro člověka triviální, tak pro stroje závisí na mnoha kritériích, které ovlivňují výsledek robota. Uvědomili jsme si důraz na kvalitu získané fotografie i jakou roli hrají kontrasty barev. Řekli bychom, že v případě hracích kostek je to naprosto klíčové.

Zkusili jsme se vydat dvěma cestami k dosažení vytyčeného cíle. Zkusili jsme cestu standardního zpracování obrazu, které ale kvůli variabilitě a častým komplikacím s jednotlivými případy se ukázala nedostačující. Druhá cesta vedoucí přes naučení neuronovou sítí jednotlivým variantám, s kterými se může potkat došla zdárného cíle s příjemnou úspěšností.

Celkově hodnotíme tento úkol jako přínosný. Ukázal nám výhody a pohodlnost využívání neuronových sítí v dnešní době. Spousta toho je naprogramováno a je možné využívat nejrozličnějších nástrojů, které již existují pro pohodlnější přípravu datasetu nebo pak posléze samotného trénování. Uvědomili jsme si, že je na tomto poli stále, co zlepšovat a objevovat.

Náš projekt by šel zlepšit a obohatit hned v několika směrech. V tuto chvíli jsme využívali pouze našich počítačů. Jistě by bylo zajímavé vyzkoušet náš program naimplementovat do robota, který by pak hlásil součet čísel. Dále by šlo zlepšit úspěšnost řešení správného čísla až už zvětšením datasetu nebo delším trénováním. Také by se dalo zapracovat na zvětšení množství případů, kdy bude schopný náš program detekovat tečky na obrazovce.

Děkujeme za možnost nahlédnout do dalšího odvětví IT a poznat možnosti a limity strojů v dnešní době.