

Summera tärningarnas valörer

Mål

Att ta fram en modell som kan deployas på enklare hårdvara, nära IoT/"edge", i syfte att avdramatisera AI för den oinvidige genom att visualisera något lekfullt och enkelt på en manick som kan hållas i handen utan koppling till en superdator i molnet.

Detta genom att analysera bilder av ett gäng tärningar på ett bord, och summera deras valörer. Mer exakt genom att låta en Nvidia Jetson Xavier NX sampla en videoström och i "realtid" (var 1-5 sek) summera tärningarnas valörer varpå summan presenteras på en skärm/logg/terminal.

Del av utmaningen och målet är att balansera prestanda, precision, svarstider etc för att hitta en modell som är snabb nog och precis nog att använda i en miljö med begränsade resurser.

Genomförande

Jag har för avsikt att använda PyTorch och Torchvision för att träna en Mask R-CNN -modell, med ResNet-50 som backbone. Denna kommer lösa både segmentering (hitta tärningarna på bordet) och klassificering (sex olika valörer).

Modellen kommer att tränas på ett dataset format enligt COCO, genom CocoDetection i Torchvision.

Jag har hittat ett annoterat dataset på Roboflow. Det innehåller ca 6000 annoterade bilder på tärningar, inkl bounding box, segmenteringsmask och klassificering. Min avsikt är att använda detta som en gedigen grund, men därutöver komplettera med några tiotal bilder som jag själv normaliserar och annoterar för att lära mig processen.

Den exakta strukturen på modellen/nätverket och kringliggande hyperparametrar etc vet jag inte i detta skede. Att laborera och utreda detta är vad jag ser som det stora jobbet i projektet.

Data

Jag ser mitt dataset som någorlunda robust och komplett. Det kommer möjligen finnas vissa edge cases där tärningar hamnar:

- tätt intill varandra, vilket jag hoppas och tror min modell kommer klara av
- lite på sniskan, vilket jag kommer tolka som felaktig input/out of scope
- (delvis) utanför bilden, vilket jag kommer tolka som felaktig input/out of scope

Några verktyg jag kommer ha till hands för att förbättra precisionen är:

- Augmentering av träningsmaterialet (förvrängning av olika slag)
- Augmentering av inferensbild (skärpa/kontrast, svart/vit)

Överväganden

Möjligen skulle jag kunna lösa stora delar av uppgiften med enklare objekt-detektion, snarare än segmentering. Jag vill dock försöka lösa problemet genom segmentering, främst för att göra min modell robust nog att hantera flera samtida tärningar som kan hamna strax intill varandra.

Jag har även övervägt en hybridlösning där man i ett första led skulle kunna använda enklare objekt-detektion och i ett nästa steg klassificera varje enskild tärning.

Nästa steg

En förlängning av detta projekt skulle kunna vara att skriva en Assistent genom OpenAIs API, och mata denna med resultatet av slagna tärningar. Genom detta skulle man kunna skapa en slags Yatzy co-pilot, och med den skulle jag möjligen ha en chans att slå min fru i Yatzy nån gång i framtiden...