



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**  
FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**DETEKCE KARET PŘI TURNAJÍCH V POKRU**  
DETECTING CARDS IN POKER TOURNAMENTS

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**VLADYSLAV KOVALETS**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. MAREK VAŠKO,**

**BRNO 2024**

## Zadání bakalářské práce



Ústav: Ústav počítačové grafiky a multimédií (UPGM)  
Student: **Kovalets Vladyslav**  
Program: Informační technologie  
Název: **Detekce karet při turnajích v pokru**  
Kategorie: Počítačové vidění  
Akademický rok: 2023/24

155626

### Zadání:

1. Seznamte se s problematikou automatického přepisu kartiček ze záběrů kamer při turnajích ve hře poker, zároveň se seznamte s moderními metodami detekce a rozpoznávání objektů v obraze a s metodami jejich tréninku.
2. Navrhněte vlastní systém pro detekci, přepis a evidenci aktuálně viditelných herních karet.
3. Najděte, případně sestavte vhodnou datovou sadu pro experimenty.
4. Iterativně vylepšujte návrh systému a experimentujte.
5. Diskutujte o budoucím vývoji.
6. Vytvořte krátké video prezentující dosažené výsledky.

### Literatura:

Podle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:  
1, 2, 3 a značně rozpracované cíle 4 a 5.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Vaško Marek, Ing.**

Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.

Datum zadání: 1.11.2023

Termín pro odevzdání: 9.5.2024

Datum schválení: 9.11.2023

## **Abstrakt**

Tato bakalářská práce se zaměřuje na vývoj pokročilého systému pro automatické rozpoznávání a evidenci herních karet z videozáznamů pokerových her. Jako základní nástroj byla zvolena technologie konvolučních neuronových sítí, konkrétně síť YOLO, který umožňuje efektivní identifikaci karet na stole i v rukou hráčů i za náročných podmínek. Práce zahrnovala vytvoření rozsáhlé datové sady pro trénování a testování detektoru karet, který dosáhl přesnosti rozpoznávání 98,7 %. Pro spolehlivou evidenci karet byl navržen algoritmus, který minimalizuje chyby detektoru a zlepšuje celkovou přesnost systému. Výsledky práce naznačují, že navržený systém má potenciál pro využití i v praxi.

## **Abstract**

This bachelor's thesis focuses on the development of an advanced system for automatic recognition and registration of playing cards from video recordings of poker games. The technology of convolutional neural networks, specifically the YOLO network, was chosen as the basic tool. It enables effective identification of cards on the table and in the hands of players even under challenging conditions. The work involved creating an extensive dataset for training and testing the card detector, which achieved a recognition accuracy of 98.7%. An algorithm was designed to minimize detector errors and improve the overall accuracy of the system. The results of the study suggest that the developed system has potential for use in practice.

## **Klíčová slova**

detekce a klasifikace herních karet, počítačové vidění, konvoluční neuronová síť, YOLO, rozpoznávání objektů, zpracování obrazu, datová sada, generování syntetických obrázků, augmentace dat, poker

## **Keywords**

card detection and classification, computer vision, convolutional neural network, YOLO, object recognition, image processing, dataset, synthetic image generation, data augmentation, poker

## **Citace**

KOVALETS, Vladyslav. *Detekce karet při turnajích v pokru*. Brno, 2024. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Marek Vaško,

# Detekce karet při turnajích v pokru

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Marka Vaška. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....  
Vladyslav Kovalets  
7. května 2024

## Poděkování

Rád bych poděkoval vedoucímu práce Ing. Marku Vaškovi za jeho cenné vedení a pravidelné konzultace, jakož i za poskytnutí přístupu k výkonnému školnímu serveru Sophie GPU. Dále bych chtěl srdečně poděkovat své matce a přítelkyni, které mi poskytly neocenitelnou morální podporu.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>5</b>
<b>2</b>	<b>Poker a klasické metody počítačového vidění</b>	<b>6</b>
2.1	Pokerové karty . . . . .	6
2.2	Výzvy detekce karet v pokeru . . . . .	7
2.3	Vyhledávání vzorů . . . . .	8
<b>3</b>	<b>Konvoluční neuronové sítě</b>	<b>11</b>
3.1	Základy konvolučních neuronových sítí . . . . .	11
3.2	Architektura CNN . . . . .	11
3.3	Faster Region-Based Convolutional Neural Networks . . . . .	15
3.4	Single Shot MultiBox Detector . . . . .	16
3.5	You Only Look Once . . . . .	17
<b>4</b>	<b>Návrh řešení</b>	<b>21</b>
4.1	Koncepce a cíle systému . . . . .	22
4.2	Trénování a testování modelu . . . . .	26
<b>5</b>	<b>Dátová sada</b>	<b>30</b>
5.1	Zdroj dat . . . . .	31
5.2	Vývoj a implementace anotačních algoritmů . . . . .	33
5.3	Následné zpracování . . . . .	35
5.4	Augmentace dat . . . . .	35
5.5	Rozdělení dat . . . . .	38
<b>6</b>	<b>Implementace</b>	<b>39</b>
<b>7</b>	<b>Výsledky experimentů</b>	<b>42</b>
7.1	Experiment 1: Vliv datových kombinací . . . . .	42
7.2	Experiment 2: Rychlosť detekce . . . . .	43
7.3	Experiment 3: Evidence herních karet . . . . .	44
<b>8</b>	<b>Záver</b>	<b>47</b>
	<b>Literatura</b>	<b>48</b>
<b>A</b>	<b>Obsah přiloženého paměťového média</b>	<b>51</b>

# Seznam obrázků

2.1	Klasický design pokerových karet. . . . .	6
2.2	Různé designy pokerových karet na příkladu dam pik. . . . .	7
2.3	Ilustrativní zobrazení výzev ovlivňujících efektivitu systémů pro rozpoznávání karet v kontextu pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1.	7
2.4	Postupný proces metody vyhledávání vzorů. Nejprve je obrázek zpracován prahováním. Následně jsou aplikovány dvě techniky a rohy seřazeny tak, aby vznikl obrázek vhodný pro závěrečnou fázi porovnávání segmentů obrazu. Pro ilustraci je na obrázku červeným čtvercem označen první roh, zeleným druhý, žlutým třetí a modrý čtvrtý. Obrázky převzaté z práce [32]. . . . .	8
3.1	Architektura konvoluční neuronové sítě. Obrázek byl adaptován z [20]. . . . .	12
3.2	Vizualizace zpracování obrázku šestky srdcí v první a druhé konvoluční vrstvě s 8 a 16 filtry, ukazující rozlišení základních tvarů a detailů symbolů. Obrázky převzaté z práce [7]. . . . .	12
3.3	Příklad vstupního obrazu a výsledné mapy příznaků po konvoluci. . . . .	13
3.4	Funkce aktivace ReLU vrací 0 pro $x < 0$ a lineární výstup se sklonem 1 pro $x > 0$ . . . . .	13
3.5	Upravená mapa příznaků po aplikaci ReLU, která zajišťuje nelinearitu v procesu zpracování obrazu tím, že odstraní všechny záporné hodnoty z mapy příznaků. . . . .	14
3.6	Rozdíl mezi maximalizačním a průměrným poolingem. . . . .	14
3.7	Zmenšená mapa příznaků výsledkem pooling operace, která snižuje rozměry mapy příznaků a zvýrazňuje nejdůležitější vlastnosti pro další vrstvy neuronové sítě. . . . .	15
3.8	Architektura Faster R-CNN. Obrázek byl adaptován z [16]. . . . .	15
3.9	Architektura SSD. Vstupní obrázek má standardní formát 300x300 pixelů s RGB kanály. Rozměry map příznaků se v průběhu zpracování sítě zmenšují (38x38x512 pixelů), zatímco jejich hloubka narůstá. Schéma ilustruje, jak sít zpracovává data v různých úrovních a rozlišuje různé velikosti objektů, čímž umožňuje efektivní detekci v reálném čase s vysokou přesností. Obrázek byl adaptován z [33]. . . . .	17
3.10	Architektura YOLOv5. Obrázek byl adaptován z [3]. . . . .	19
3.11	Pět základních variant velikostí modelu YOLOv5. Obrázek byl převzat z [34].	19
3.12	Optimalizace kotvících boxů pomocí metody k-průměrů v YOLOv5. Obrázek byl adaptován z [8]. . . . .	20
4.1	Proces řešení pro detekci, klasifikaci a evidenci pokerových karet. Začíná sběrem dat, následuje trénování modelu a poté aplikace modelu na videa z pokerových turnajů a zpracování získaných dat. . . . .	21

4.2	Pohled z kamery shora na pokerový stůl, kde krupíér rozkládá karty. Zdroj je uveden v Tabulce 5.1. . . . .	22
4.3	Pohled z kamery zabudované do hran stolu, kde hráč se dívá na své karty. Zdroj je uveden v Tabulce 5.1. . . . .	23
4.4	Detekce karet pomocí identifikace jejich rohů. Zeleným obdélníkem je vyznačena oblast obsahující veškeré informace o kartě. Zdroje jsou uvedeny v Tabulce 5.1. . . . .	23
4.5	Příklad snímku, kde je nutné použít oblast zájmu. Karty ležící mimo tuto oblast nejsou rozpoznávány, jelikož jsou považovány za nepotřebné. Zdroj je uveden v Tabulce 5.1. . . . .	24
4.6	Fragment videa, jehož JSON soubor je uveden ve Výpisu 4.2. Zdroj je uveden v Tabulce 5.1. . . . .	25
4.7	Míra překrytí mezi předdefinovanými a detekovanými obalovými obdélníky v závislosti na jejich pozici. Čím vyšší je hodnota IoU, tím větší je shoda mezi očekávanou a detekovanou polohou. Obrázek byl adaptován z [17]. . . . .	27
5.1	Příklady synteticky generovaných obrázků pokerových karet, získaných z datové sady na platformě Roboflow. . . . .	30
5.2	Příklady obrázků pokerových karet, umístěných na herním stole, získaných z videozáZNAMŮ pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1. . .	31
5.3	Příklady obrázků pokerových karet, držených v rukou hráčů, získaných z videozáZNAMŮ pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1. . . . .	31
5.4	První fotografie je originální snímek, zatímco napravo od něj jsou umístěny obrázky, které byly generovány na jeho základě. . . . .	32
5.5	Příklady synteticky generovaných obrázků pokerových karet, které jsem vytvořil pro rozšíření datové sady. . . . .	33
5.6	Ukázka přesného zachycení hodnot a barev karet pomocí obalových obdélníků. Zdroje jsou uvedeny v Tabulce 5.1. . . . .	34
5.7	Příklady různých augmentačních technik aplikovaných na stejný obrázek (původní obrázek je zobrazen na Obrázku 5.2 vpravo). Zleva doprava: obrázek po aplikaci rotace a úpravy odstínu a sytosti barev, obrázek po deformaci a posunutí, obrázek po aplikaci černobílého filtru a zmenšení. Zdroj je uveden v Tabulce 5.1. . . . .	36
5.8	Příklady různých augmentačních technik aplikovaných na stejný obrázek (původní obrázek je zobrazen na Obrázku 5.3 vlevo). Zleva doprava: obrázek po aplikaci rotace a přidání zrnitosti, obrázek po aplikaci rozostření a změny perspektivy, obrázek po aplikaci posterizace a deformace. Zdroj je uveden v Tabulce 5.1. . . . .	37
5.9	Ukázka augmentace obrázků při tréninku detekčního modelu YOLOv5. Na obrázku je zobrazeno osm augmentovaných snímků, které zahrnují mozaikovou augmentaci, modifikace barev a další techniky. . . . .	38
6.1	Ukázka anotace obrázku na platformě Roboflow. . . . .	39
6.2	Vytvořený zobrazovací nástroj pro zobrazení detekovaných karet na herním stole a v rukách hráčů. Snímek z videa demonstruje ikony rozpoznaných karet, které jsou viditelné v levém horním rohu. Na snímku je také uvedeno jeho číslo. Zdroj je uveden v Tabulce 5.1. . . . .	40

7.1	Graf porovnávající frekvenci zpracování snímků ( <i>angl.</i> FPS) a mAP50 modelů YOLOv5 v rámci Experimentu 2. Označení: X (YOLOv5x), L (YOLOv5l), M (YOLOv5m), S (YOLOv5s), N (YOLOv5n). . . . .	44
7.2	Výseky z videa zachycující karty různých hráčů. Zdroje jsou uvedeny v Tabulce 5.1. . . . .	45
7.3	Výsek z videa, zachycující rozkládání karet krupiérem na herním stole. Zdroj je uveden v Tabulce 5.1. . . . .	45

# Kapitola 1

## Úvod

Svět pokeru se v posledních letech stává čím dál více propojeným s moderními technologiemi. Pokročilé metody zpracování obrazu otevírají nové možnosti pro vylepšení zážitku pro všechny zúčastněné. Jednou z nejzajímavějších oblastí je rozpoznávání karet v reálném čase, které by mohlo revolučním způsobem změnit způsob, jakým se poker sleduje a analyzuje.

Tato práce se zaměřuje na vývoj systému pro rozpoznávání karet v pokerových hrách. Systém by měl být schopen detektovat, přepisovat a zaznamenávat herní karty, které se nacházejí jak na stole, tak v rukou hráčů. Využití systému rozpoznávání karet by mělo pozitivní dopad na všechny aspekty pokerových turnajů, od usnadnění práce organizátorům až po zpříjemnění herního zážitku pro hráče a diváky.

Poker je hra kombinující dovednosti, strategii a štěstí, která se vždy vyvijela v souladu s technologickým pokrokem. V minulosti byly používány různé metody pro sledování karet, od manuálního zapisování až po jednoduché systémy rozpoznávání obrazu. Tato práce představuje další krok vpřed díky použití metod hlubokého učení, které slibují výrazně vyšší přesnost a efektivitu.

Hlavním cílem této práce bylo vyvinout robustní a efektivní systém pro automatické rozpoznávání a identifikaci karet při pokerových turnajích. Systém využívá moderní metody zpracování obrazu a byl navržen tak, aby fungoval i v náročných podmínkách, které jsou pro pokerové turnaje typické. Mezi tyto podmínky patří různé osvětlení, překrývání karet hráči a rychlé tempo hry.

Pro dosažení tohoto cíle byla sestavena rozsáhlá datová sada, která obsahuje obrázky pokerových her s anotovanými kartami. Při vývoji modelu pro rozpoznávání karet byly využity konvoluční neuronové sítě, které jsou široce uznávány pro svou přesnost v zpracování obrazu. Tento model byl pečlivě trénován na zmíněné datové sadě, jež byla průběžně rozšiřována a upravována, aby bylo možné zajistit optimální funkčnost modelu v různých situacích. Výsledkem je model s vysokou přesností rozpoznávání karet, dosahující 98,7 % v podmínkách pokerových turnajů.

I když je absolutní bezchybnost modelu v praxi téměř nedosažitelná, byl zaveden algoritmus, který minimalizuje chyby v konečné evidenci karet. Po dokončení analýzy pokerového zápasu tak výsledná evidence odráží vysoko přesné a spolehlivé údaje o kartách.

## Kapitola 2

# Poker a klasické metody počítačového vidění

Hlavním cílem této kapitoly je prezentovat klasickou metodu detekce karet a prozkoumat možnosti využití této metody pro identifikaci karet během pokerových turnajů. V úvodu kapitoly je poskytnuto základní informace o pokerových kartách, včetně jejich typů a designových variant. Dále se analyzují výzvy, kterým systémy detekce karet čelí, včetně světelních podmínek a dynamiky turnajového prostředí. Detailně se popíše klasická metoda vyhledávání vzorů a techniky, které jsou s ní spojené. Kapitola vrcholí diskuzí o současných omezeních těchto metod a zdůrazňuje nutnost vyvíjet robustnější a adaptabilnější systém, který je klíčový pro zlepšení identifikace karet a tím i kvality pokerových turnajů.

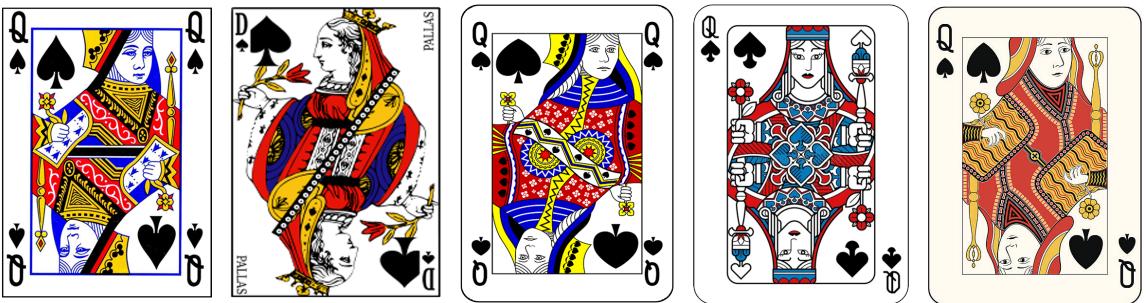
### 2.1 Pokerové karty

Poker je karetní hra, která tradičně využívá francouzský balíček karet. Tato specifická sada karet hráje klíčovou roli v herním procesu, jelikož představuje komplexní systém barev a hodnot. Pro plynulý průběh hry je nezbytné tyto barvy a hodnoty správně rozpoznávat a třídit.

**Typy pokerových karet** se rozlišují dle několika kritérií, z nichž nejzákladnější je rozdělení podle barvy a hodnoty. Standardní balíček obsahuje 52 karet rozdělených do čtyř barev: srdce, káry, piky a kříže. Každá barva zahrnuje 13 karet: čísla od 2 do 10 a figurky – kluk, královna, král a eso. Tento systém nejen usnadňuje identifikaci a porovnávání karet, ale také představuje základní prvek strategie v pokeru.



Obrázek 2.1: Klasický design pokerových karet.



Obrázek 2.2: Různé designy pokerových karet na příkladu dam pik.

Design pokerových karet se v průběhu let vyvíjel a dnes existuje mnoho variant, od klasických designů uznávaných všeobecně, představených na Obrázku 2.1, až po specializované edice vytvořené pro sběratele nebo konkrétní turnaje, jak je vidět na Obrázku 2.2. Ačkoli množství těchto designů zdůrazňuje estetickou hodnotu hry, může to představovat významnou výzvu pro detektory karet. Každý design může mít jiné rozmístění hodnot, odlišné barvy a dokonce i unikátní grafické prvky, které komplikují proces rozpoznávání.

## 2.2 Výzvy detekce karet v pokeru

Zdánlivě triviální úkol automatického rozpoznávání karet ve skutečnosti skrývá celou řadu komplexních výzev. Ilustrace těchto výzev je názorně prezentována na Obrázku 2.3, kde jsou zobrazeny klíčové aspekty, s nimiž se musí detekční systémy vypořádat.



(a) Světelné podmínky.

(b) Dynamické prostředí.



(c) Zakrývání karet.



(d) Variabilita umístění.

Obrázek 2.3: Ilustrativní zobrazení výzev ovlivňujících efektivitu systémů pro rozpoznávání karet v kontextu pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1.

**Světelné podmínky.** Pokerové turnaje se konají v rozmanitých prostředích, od malých neformálních herních zón až po velkolepé profesionální turnaje s televizním přenosem. Každé prostředí přináší specifické světelné podmínky, pozadí a další faktory ovlivňující kvalitu obrazu karet.

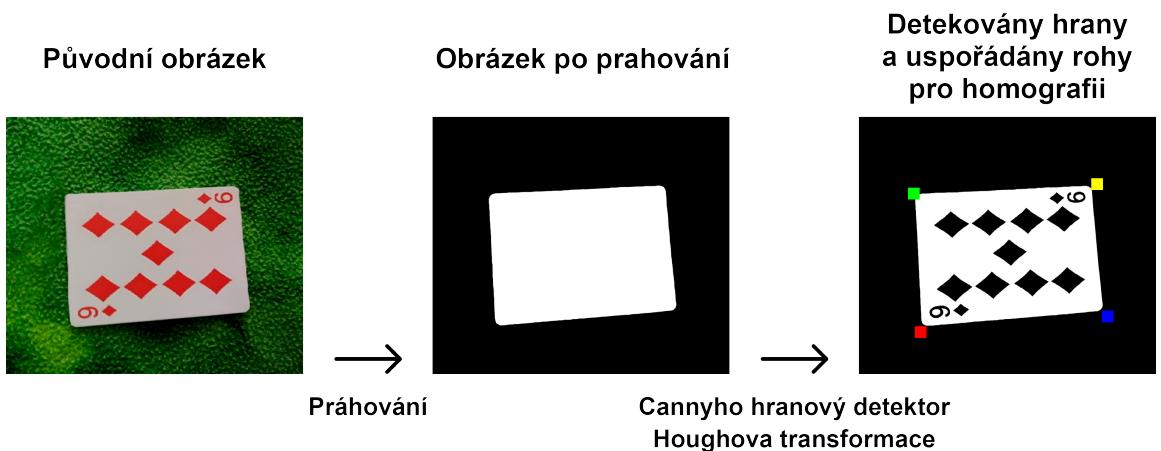
**Zakrývání karet.** Hráči běžně zakrývají karty rukama, aby je skryli před soupeři. Toto zakrytí může být částečné nebo úplné a měnit se v průběhu hry. Detekční systém musí zvládat práci s částečně viditelnými kartami.

**Dynamické prostředí.** Vzhledem k rychlému tempu pokerových her je zásadní, aby systém dokázal karty detektovat rychle a v reálném čase. Jakékoli zpoždění v detekci by mohlo negativně ovlivnit plynulost hry.

**Variabilita umístění karet.** Karty se na stole nacházejí v rozličných pozicích a pod různými úhly. Detekční systém musí být schopen identifikovat karty bez ohledu na jejich umístění a orientaci.

### 2.3 Vyhledávání vzorů

Výzvy spojené s detekcí karet v pokeru zdůrazňují nutnost vytvoření robustního a adaptabilního systému, schopného rychle a přesně fungovat v dynamickém a nepředvídatelném prostředí pokerových turnajů. Vzhledem k prvním krokům při řešení tohoto náročného úkolu se v této sekci podrobň rozebírá klasická metoda vyhledávání vzorů a její techniky, jejichž aplikace a teoretické základy vycházejí z práce o detekci a identifikaci hracích karet [32]. Postup metody je dále ilustrován Obrázkem 2.4.



Obrázek 2.4: Postupný proces metody vyhledávání vzorů. Nejprve je obrázek zpracován prahováním. Následně jsou aplikovány dvě techniky a rohy seřazeny tak, aby vznikl obrázek vhodný pro závěrečnou fázi porovnávání segmentů obrazu. Pro ilustraci je na obrázku červeným čtvercem označen první roh, zeleným druhý, žlutým třetí a modrý čtvrtý. Obrázky převzaté z práce [32].

Metoda vyhledávání vzorů porovnává segmenty obrazu s předdefinovanými vzory pro nalezení shod. Proces zahrnuje tréninkové a testovací sady, kde tréninková sada obsahuje

obrázky upravené prahováním pro vytvoření vzorů. Identifikace objektů pak probíhá porovnáním těchto vzorů s testovacími obrázky, přičemž se vypočítává skóre shody pro určení nejbližší odpovídajícího vzoru.

**Prahování obrazu** je často výchozím bodem při zpracování digitálních obrázků. V tomto procesu je barevný obraz obvykle ve formátu RGB, kde každý pixel je definován třemi hodnotami: červenou  $r$ , zelenou  $g$  a modrou  $b$ . Pro zjednodušení zpracování obrazu se barvy převádějí na šedotónový obraz průměrováním RGB hodnot

$$Intenzita = \frac{r + g + b}{3}, \quad (2.1)$$

kde  $r, g, b$  jsou hodnoty červené, zelené a modré složky pixelu v barevném obrazu.

Poté se používá technika prahování, při které se porovnávají intenzity pixelů s prahovou hodnotou. Výsledkem je binární obraz: pixely nad prahem mají hodnotu „1“, ostatní „0“.

Existují dva hlavní typy prahování:

- **Globální** – Aplikuje na celý obraz jednotnou prahovou hodnotu.
- **Adaptivní** – Upravuje prahovou hodnotu podle lokálních charakteristik obrazu.

Pro automatické stanovení prahové hodnoty se často používá Otsuho metoda [23], analyzující histogram intenzity a optimalizující rozptyl mezi třídami pro lepší oddělení popředí od pozadí.

**Cannyho hranový detektor** [6] je důležitým nástrojem pro detekci hran v digitálních obrazech, známý svou přesností a odolností proti šumu. Základní kroky algoritmu zahrnují:

1. **Eliminace šumu** pomocí Gaussova filtru.
2. **Výpočet gradientu** jasu v obou, vertikálním i horizontálním, směrech.
3. **Potlačení nemaximálních hodnot** pro zvýšení ostrosti hran.
4. **Dvojité prahování s hysterézí** pro spojení hrany a zajištění jejich kontinuity.

Algoritmus je adaptabilní na různé obrazové podmínky a spolehlivě identifikuje hranice objektů, což mu zajistilo preferovanou pozici mezi algoritmy pro detekci hran dodnes.

**Houghova transformace** [9] se po detekci hran pomocí Cannyho detektoru používá k detekci přímek v obrazu, což umožňuje přesné určení rohů objektů, jako jsou karty. Tato metoda pracuje v parametrickém prostoru s parametry  $\rho$  a  $\theta$ , které reprezentují vzdálenost a úhel přímky v polárním souřadnicovém systému. Hlavní součástí transformace je rovnice

$$\rho = x \cos(\theta) + y \sin(\theta), \quad (2.2)$$

kde  $\rho$  je vzdálenost od původu k přímce,  $\theta$  je úhel, pod kterým je přímka nakloněna k ose  $x$ , a  $x, y$  jsou souřadnice bodů na hraně objektu. Rovnice tedy propojuje tyto body s přímkami v parametrickém prostoru, což umožňuje identifikaci a lokalizaci lineárních segmentů v obrazu.

Transformace vytváří akumulační matici pro identifikaci dominantních přímek, které reprezentují skutečné hranice objektů. Tyto přímkы se pak převádějí zpět do kartézského souřadnicového systému, umožňující výpočet průsečíků a přesné určení rohů objektů.

Pro zvýšení přesnosti je důležité rohy seřadit, aby se předešlo deformaci při výpočtu transformace. Po určení a seřazení rohů se vytvoří transformační matice pro nalezení homografie mezi vzorem a analyzovaným obrazem, což umožňuje ověřit přesnost a kvalitu detekce.

## Možnosti a omezení aplikace

V rámci posuzování možností aplikace metody vyhledávání vzorů je důležité zmínit, že dle Snyderova výzkumu [32] proces identifikace každé karty trvá průměrně minutu. To je způsobeno nutností opakovaného hodnocení pro každou ze 52 karet, což značně omezuje rychlosť procesu a snižuje jeho praktickou použitelnost.

Podle tohoto výzkumu bylo rovněž prokázáno vysoká úroveň přesnosti algoritmu, dosahující 98 %, avšak testování probíhalo na poměrně malém souboru dat (pouze 52 obrazků). Tento výzkum rovněž zkoumal odolnost algoritmu vůči perspektivnímu zkreslení a odhalil, že algoritmus je schopen identifikovat karty i při mírném zkreslení, avšak při větším zkreslení docházelo k chybám způsobeným nesprávnou identifikací relevantních oblastí karty, což snižovalo přesnost shody. Autor výzkumu navíc prováděl testování za ideálních podmínek. To zahrnovalo optimální osvětlení, nepohyblivé a nepřekryté karty bez perspektivního zkreslení. V praxi se však algoritmus může setkat s méně ideálními scénáři, které jsou popsány v sekci 2.2.

Jiný výzkum [15] naznačuje, že přesnost klasifikace se může lišit v závislosti na výběru předdefinovaných vzorů a nastavení prahu podobnosti. Nejvyšší dosažená přesnost byla 81.81 % při specifických nastaveních, které nemusí být vždy optimální pro veškeré herní situace.

Mezi hlavní nevýhody této metody patří časová náročnost, její citlivost na různé osvětlení a deformace karet, jako jsou ohýbání, rotace nebo naklonění. To poukazuje na potřebu vyvíjet nové, robustnější a flexibilnější systémy. Například, systém založený na pokročilé kombinaci strojového vidění a umělé inteligence by mohl být navržen tak, aby byl schopen adaptovat se na různé světelné podmínky a dynamicky reagovat na fyzické změny karet. Tato technologie by mohla zahrnovat algoritmy pro hluboké učení, které by se učily z velkého množství obrazových dat a stále se zdokonalovaly. Takový systém by značně zlepšil přesnost a spolehlivost identifikace karet a zvýšil celkovou kvalitu průběhu pokerových turnajů.

## Kapitola 3

# Konvoluční neuronové sítě

Konvoluční neuronové sítě (*angl. CNN*<sup>1</sup>) [22] představují jedny z nejpoužívanějších nástrojů v oblasti počítačového vidění, a to především v kontextu rozpoznávání obrazů. Tato kapitola se podrobně zaměří na svět CNN, jejich specifické vlastnosti a hlavní přednosti oproti tradičním metodám zpracování obrazu, které se vyvíjely v průběhu posledních desetiletí. Zvláštní pozornost bude věnována vysvětlení, proč je tento přístup zvláště vhodný pro detekci hracích karet.

### 3.1 Základy konvolučních neuronových sítí

Neuronové sítě [30] se skládají z mnoha propojených neuronových jednotek, které si mezi sebou předávají informace. Tyto jednotky jsou inspirovány fungováním lidského mozku a umožňují sítím rozpoznávat složité vzory v datech.

Trénink neuronových sítí probíhá na základě sady příkladů s již známými výsledky. Sít se učí z těchto příkladů a postupně zlepšuje svou přesnost v predikci správného výstupu, podobně jako se učíme z vlastních zkušeností.

Speciální typ neuronové sítě, nazývaný konvoluční neuronová síť, je obzvláště vhodný pro úkoly související s viděním, inspirovaný částí mozku, která zpracovává vizuální informace. Tyto sítě jsou navrženy tak, aby rozpoznávaly vzory v malých oblastech vstupních dat, podobně jako naše zraková soustava se zaměřuje na specifické části toho, co vidíme, abychom pochopili celý obrázek.

### 3.2 Architektura CNN

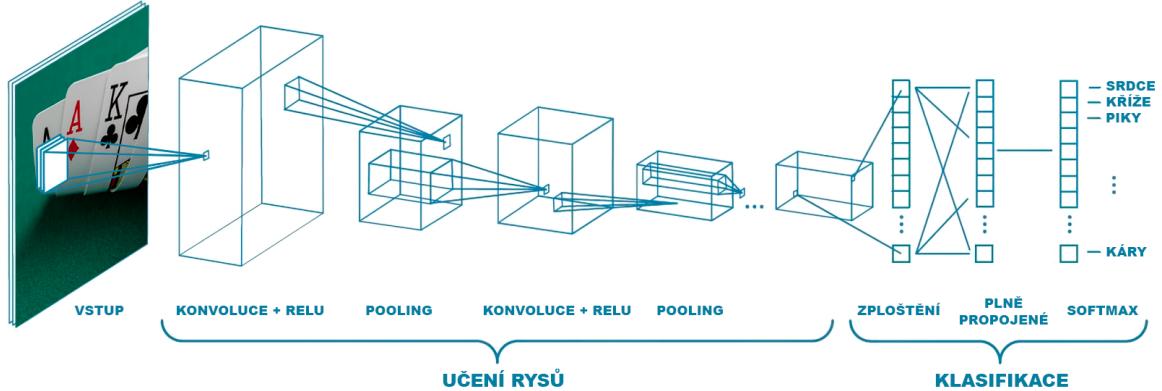
Konvoluční neuronová síť se skládá z několika vrstev, které postupně extrahují a transformují informace z vstupního obrázku. Proces zpracování dat je zobrazen na Obrázku 3.1. Následující odstavce se zaměří na detailní popis těchto vrstev a jejich funkcí.

**Konvoluční vrstvy** v neuronových sítích slouží k vyhledávání a získávání různých příznaků ze vstupních obrázků. Pro tento účel používají různé filtry, které se během trénovacího procesu učí a přizpůsobují. Tyto filtry se aplikují na celý obrázek, kde konvolují s daty na obrázku a vypočítávají tak relevantní příznaky. Každý filtr je zaměřený na odlišný příznak a všechny společně tvoří kolekci map příznaků, které jsou dále použity v síti. Výsledkem

---

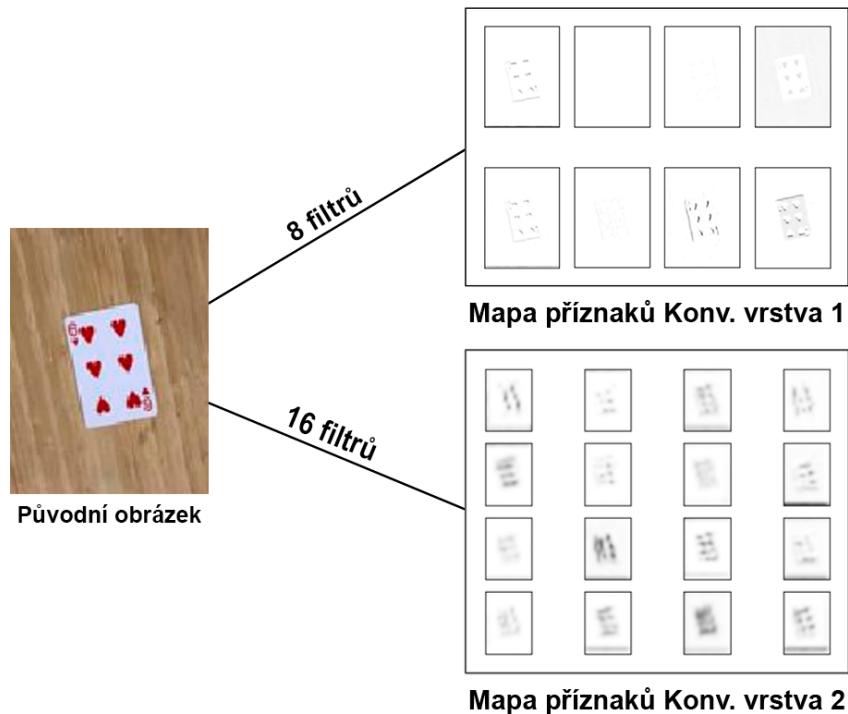
<sup>1</sup>CNN – *Convolutional Neural Network* (konvoluční neuronová síť).

práce konvolučních vrstev jsou tedy tyto mapy příznaků, které představují rozpoznané charakteristiky původních obrázků.



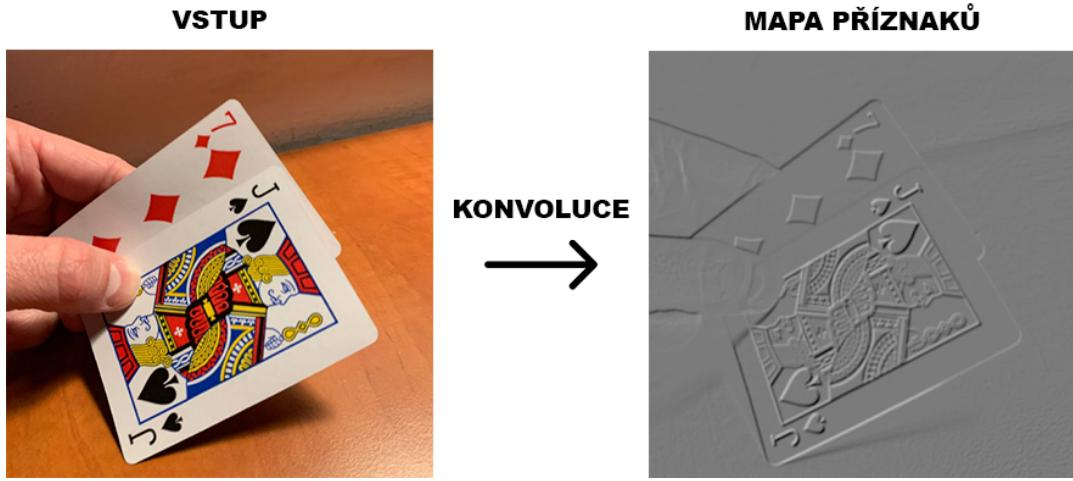
Obrázek 3.1: Architektura konvoluční neuronové sítě. Obrázek byl adaptován z [20].

Následující Obrázek 3.2 demonstroval princip fungování konvolučních vrstev na příkladu karty šestky srdečí. Ilustrace zobrazuje původní obrázek karty a jeho postupné procházení konvolučními vrstvami. První vrstva s 8 filtry se zaměřuje na záchyt základních struktur, jako jsou hrany a obrysy karty, zatímco druhá vrstva s 16 filtry odhaluje jemnější detaily a specifika vzorů na kartě. Tímto způsobem lze vidět, jak různé filtry extrahuji rozličné úrovně informací, což je klíčové pro úspěšnou klasifikaci a rozpoznavání složitých vzorců v obrazech.



Obrázek 3.2: Vizualizace zpracování obrázku šestky srdečí v první a druhé konvoluční vrstvě s 8 a 16 filtry, ukazující rozlišení základních tvarů a detailů symbolů. Obrázky převzaté z práce [7].

Na těchto Obrázcích 3.3, 3.5 a 3.7 je zobrazena vizualizace procesu aplikace operací v konvoluční neuronové síti. To umožňují sledovat, jaké příznaky neuronová síť na jednotlivých vrstvách rozpoznává a jak tyto prvky ovlivňují finální rozpoznávání vzorů.



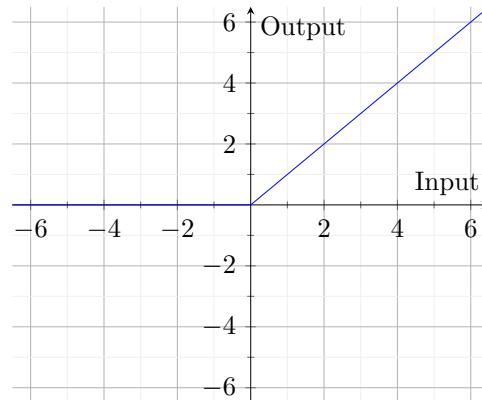
Obrázek 3.3: Příklad vstupního obrazu a výsledné mapy příznaků po konvoluci.

Obrázek 3.3 zobrazuje aplikaci konvoluce na vstupní obraz. Po aplikaci konvoluce vzniká „Mapa příznaků“, která reprezentuje vstupní obraz s zvýrazněnými relevantními vlastnostmi.

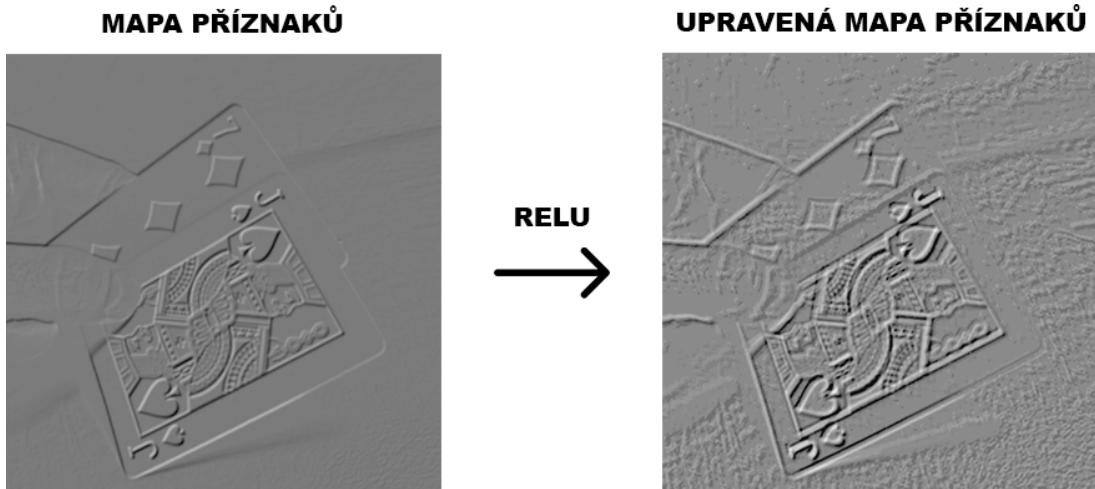
**Vrstva Rectified Linear Unit (ReLU)** [1] je funkce používaná v mapách příznaků po jejich extrakci. Funkce je definována velmi jednoduše: pro záporné hodnoty vrací nulu, zatímco pro nezáporné hodnoty vrací stejnou hodnotu. Tuto funkčnost lze formálně zapsat vztahem

$$f(x) = \max(0, x). \quad (3.1)$$

Na Obrázku 3.4 je demonstrována tato funkce v praxi, kde výstup pro záporné hodnoty je nula a pro kladné hodnoty se zvyšuje lineárně s  $x$ . ReLU tedy přináší do neuronové sítě důležitou nelinearitu, která je klíčová pro modelování komplexních vztahů mezi daty.



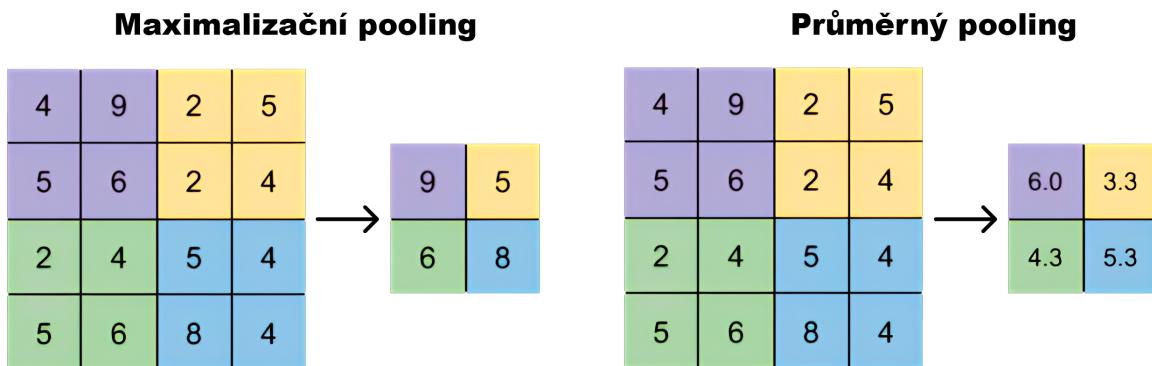
Obrázek 3.4: Funkce aktivace ReLU vrací 0 pro  $x < 0$  a lineární výstup se sklonem 1 pro  $x > 0$ .



Obrázek 3.5: Upravená mapa příznaků po aplikaci ReLU, která zajišťuje nelinearitu v procesu zpracování obrazu tím, že odstraní všechny záporné hodnoty z mapy příznaků.

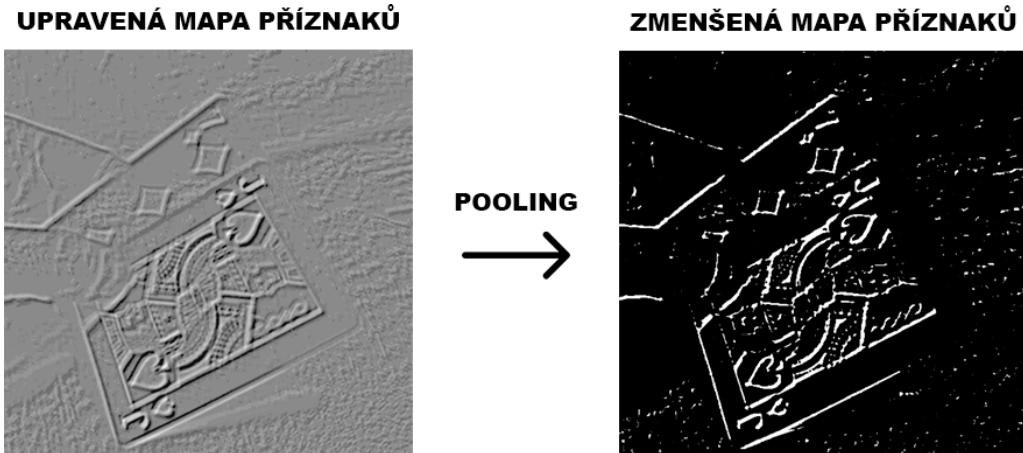
Další Obrázek 3.5 ukazuje aplikaci aktivační funkce ReLU, která dále upravuje mapu příznaků odstraněním negativních hodnot a ponecháním pouze kladných. Výsledkem je „upravená mapa příznaků“, která je připravena pro další vrstvy konvoluční neuronové sítě.

**Pooling [22]** snižuje velikost map příznaků, čímž zajišťuje menší citlivost na polohu prvků a zjednodušuje model. Existují různé typy: *průměrný pooling* bere průměr hodnot, *maximalizační pooling* bere nejvyšší hodnotu z oblasti a *globální maximalizační pooling* vybírá nejvyšší hodnotu z celého vstupu (rozdíl mezi maximalizačním a průměrným poolingem je zobrazen na Obrázku 3.6). Využití pooling operací pomáhá v redukci rozměrů dat při zachování důležitých příznaků, což může vést k efektivnějšímu a rychlejšímu učení neuronové sítě a lepší generalizaci na neviděných datech.



Obrázek 3.6: Rozdíl mezi maximalizačním a průměrným poolingem.

Na dalším Obrázku 3.7 je znázorněn maximalizační pooling, který redukuje rozložení mapy příznaků a zachovává pouze nejvýraznější hodnoty v určitých oblastech. Tímto krokem vzniká „zmenšená mapa příznaků“, která je efektivnější pro další zpracování a snižuje výpočetní náročnost.



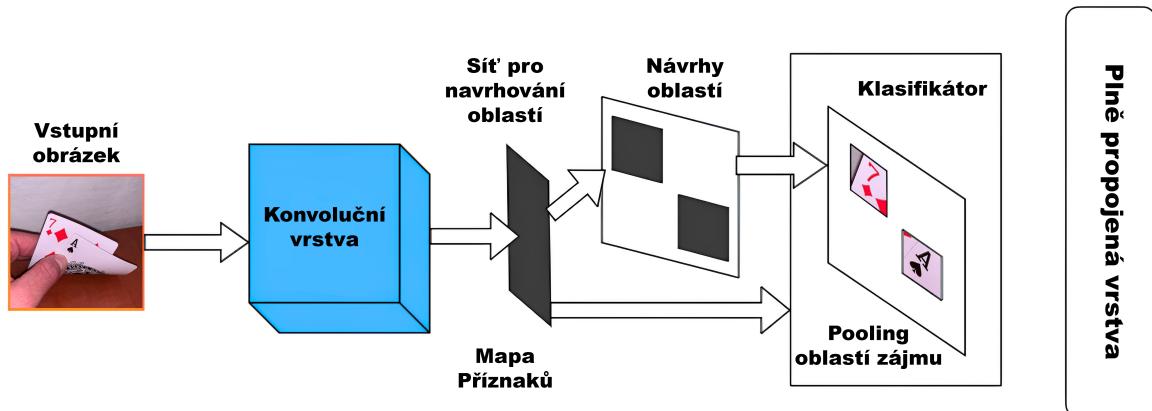
Obrázek 3.7: Zmenšená mapa příznaků výsledkem pooling operace, která snižuje rozměry mapy příznaků a zvýrazňuje nejdůležitější vlastnosti pro další vrstvy neuronové sítě.

**Plně propojené vrstvy [36]** slouží k vytvoření konečného výsledku nebo predikce. Před tímto krokem však prochází výstup z konvolučních a pooling vrstev procesem zploštování, kde se vícedimenziorní mapy příznaků transformují na jednorozměrný vektor. Tento zploštěný vektor je následně předán do sady neuronů, které jsou mezi sebou plně propojeny. Neurony pak provádějí skalární součin těchto vstupů se svými váhami, což umožňuje síti naučit se optimální váhy pro různé účely, jako je klasifikace vstupních dat nebo regrese parametrů pro určení polohy a rozměrů objektů v obraze. Po výpočtu se použije aktivační funkce, která rozhoduje o aktivaci výstupů neuronů, zavádějíc současně nelinearitu do procesu.

Poslední vrstva typicky využívá *softmax*, což je aktivační funkce normalizující výstupní hodnoty na pravděpodobnosti jednotlivých tříd. Třída s nejvyšší pravděpodobností je pak vybrána jako predikovaná, což dává přehled o jistotě klasifikace vstupu.

### 3.3 Faster Region-Based Convolutional Neural Networks

Faster Region-Based Convolutional Neural Networks (Faster R-CNN) [28] představuje významný pokrok v oblasti detekce objektů. Model byl vyvinut v roce 2015 a stal se základem pro mnoho moderních systémů detekce objektů.



Obrázek 3.8: Architektura Faster R-CNN. Obrázek byl adaptován z [16].

Obrázek 3.8 znázorňuje architekturu Faster R-CNN, která se skládá z vrstev, které postupně zpracovávají vstupní obraz a provádějí klasifikaci objektů.

## Revoluční aspekty

**Sít pro navrhování oblastí (RPN<sup>2</sup>)** automaticky generuje oblasti zájmu z obrázků, čímž umožňuje modelu soustředit se na relevantní části a zefektivňuje detekci.

**Dvoufázový proces** obsahuje dvě klíčové komponenty, které zvyšují přesnost a efektivitu detekce objektů. V první fázi RPN analyzuje vstupní obraz a generuje návrhy oblastí, které mohou obsahovat objekty. Tyto návrhy se extrahuji na základě příznaků získaných z předchozích konvolučních vrstev, což umožňuje lokalizaci objektů nezávisle na jejich velikosti a pozici.

Ve druhé fázi je použit Fast R-CNN<sup>3</sup> [12], který tyto návrhy oblastí dále zpracovává za účelem klasifikace a přesné lokalizace objektů. Fast R-CNN přijímá zploštěné výstupy RPN a aplikuje na ně další konvoluční a plně propojené vrstvy.

**Obalové obdélníky** slouží jako referenční body pro generování návrhů oblastí. Model využívá tyto obdélníky různých velikostí a tvarů, které následně upravuje tak, aby optimalizoval předpovědi polohy a třídy objektů v obraze.

**Trénování od začátku do konce** umožňuje trénovat celý systém najednou, což vede ke zvýšení efektivity a kvality výsledků.

Přestože Faster R-CNN přináší tyto inovace a přednosti, pro specifický případ detekce karet během turnaje se může ukázat jako méně vhodný z několika důvodů. Výpočetní náročnost modelu je stále významná, i když je nižší než u jeho předchůdců, což může být problematické v prostředí, kde je potřeba rychle a v reálném čase zpracovávat velké množství obrazových dat. Navíc, i když Faster R-CNN efektivně detekuje a klasifikuje širokou škálu objektů, může mít potíže s detekcí malých objektů, jako jsou rohy hracích karet, v komplexních scénách. Z těchto důvodů je Faster R-CNN nevhodnou volbou pro detekci karet během turnajů v poker.

## 3.4 Single Shot MultiBox Detector

Výzkum publikovaný koncem roku 2016 pod názvem „SSD: Single Shot MultiBox Detector“ C. Szegedy et al. [19], znamenal zásadní posun v oblasti detekce objektů v reálném čase. Vyniká svou efektivitou a přesností. Na rozdíl od Faster R-CNN, SSD<sup>4</sup> umožňuje detekci objektů v jediném kroku bez nutnosti předchozího navrhování oblastí. To značně zkracuje dobu zpracování při zachování vysoké přesnosti.

Na přiloženém Obrázku 3.9 je zobrazena architektura SSD, která demonstriuje, jak tento systém využívá upravenou verzi konvoluční neuronové sítě VGG16<sup>5</sup> [31] k extrakci rysů z obrázků a následně aplikuje konvoluční filtry pro produkci detekcí, klasifikací a pravděpodobnosti přímo na těchto extrahovaných rysech. Díky implementaci algoritmu pro více

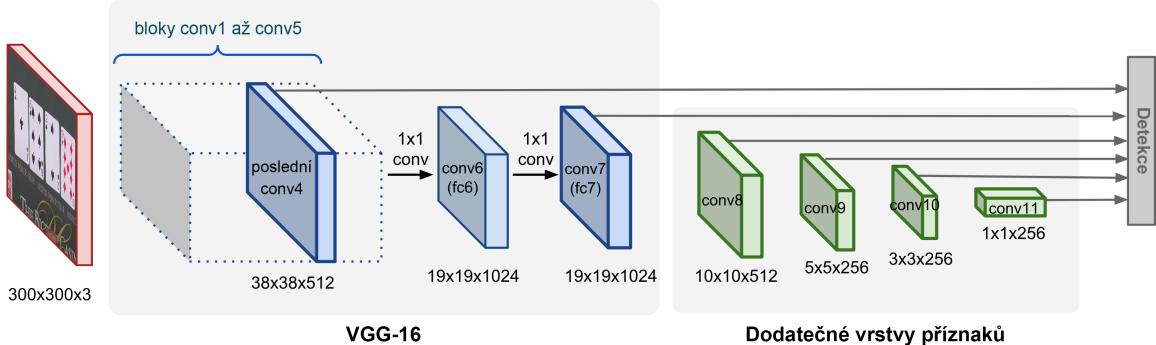
<sup>2</sup>RPN – *Region Proposal Network* (sít pro navrhování oblastí).

<sup>3</sup>Fast R-CNN – *Fast Region-Based Convolutional Neural Networks* (rychlé regionální konvoluční neuronová síť).

<sup>4</sup>SSD – *Single Shot MultiBox Detector* (jednoetapový detektor s více obdélníky).

<sup>5</sup>VGG-16 – *Visual Geometry Group-16* (vizuální geometrická skupina z 16 konvolučních vrstev).

obdélníků (*angl.* Multibox) [11], který integruje vrstvy s různou mírou rozlišení a sadu obalových obdélníků, je systém schopen detekovat objekty různých velikostí s vysokou přesností i v případě obrázků s nižším rozlišením.



Obrázek 3.9: Architektura SSD. Vstupní obrázek má standardní formát 300x300 pixelů s RGB kanály. Rozměry map příznaků se v průběhu zpracování síť zmenšují (38x38x512 pixelů), zatímco jejich hloubka narůstá. Schéma ilustruje, jak síť zpracovává data v různých úrovních a rozšiřuje různé velikosti objektů, čímž umožňuje efektivní detekci v reálném čase s vysokou přesností. Obrázek byl adaptován z [33].

Hlavními výhodami SSD jsou schopnost provádět detekci v reálném čase, což je klíčové pro aplikace vyžadující okamžitou odezvu, jako jsou autonomní vozidla nebo bezpečnostní systémy, a efektivní spojení klasifikačních a lokalizačních procesů do jediného kroku. Díky tomu je zpracování rychlejší.

I když SSD přináší značné vylepšení v detekci objektů v reálném čase, což by teoreticky mohlo být výhodné pro detekci pokerových karet během turnajů, jeho využití v tomto specifickém kontextu narází na podobné výzvy, které jsou zmíněny u Faster R-CNN. Přestože SSD není tak výpočetně náročný a nabízí rychlejší zpracování obrazu, jeho schopnost detekovat malé objekty, jako jsou rohy karet s barvou a hodnotou, může být limitována.

### 3.5 You Only Look Once

You Only Look Once (YOLO) [26] je algoritmus pro detekci objektů v reálném čase, vyvinutý Josephem Redmonem v roce 2016. Na rozdíl od dvoufázových algoritmů detekce, YOLO, stejně jako SSD, provádí klasifikaci a lokalizaci objektů v jednom kroku přímo na celém obrázku, čímž umožňuje výrazně rychlejší zpracování.

Poslední verzí, kterou Redmon v roce 2018 vydal, je YOLOv3 [27]. Nicméně, v dubnu 2020 další výzkumníci publikovali algoritmus YOLOv4 [4], který rozvíjí původní koncepty a přináší další vylepšení v efektivitě a přesnosti detekce. Pouze dva měsíce poté, v červnu 2020, se nečekaně objevila YOLOv5 [34].

#### YOLOv4

Hlavní zdokonalení oproti předchozím verzím spočívá ve využití nových technik a konfigurací, které zvyšují efektivitu a výkon bez potřeby extrémně výkonného hardwaru.

**Architektura YOLOv4** je založena na třech základních komponentách: základní vrstva (*angl.* backbone), propojovací vrstvy (*angl.* neck) a vrcholové vrstvy (*angl.* head). Základní

vrstva využívá síť CSPDarknet53 [35], která je optimalizována pro rychlé a přesné zpracování obrazu. Tato vrstva je klíčová pro extrakci rysů z vstupních obrazových dat. Propojovací vrstvy jsou navrženy tak, aby efektivně kombinovaly a rafinovaly rysy získané ze základní vrstvy. Pro tyto účely je použit blok Spatial Pyramid Pooling (SPP) [13] a struktura Path Aggregation Network (PAN) [18], které umožňují modelu zachytit podstatné informace na různých měřítkách a zlepšit integraci mezi vrstvami. Vrcholové vrstvy jsou odpovědné za detekci a klasifikaci objektů. Tato část modelu pracuje s daty přicházejícími z propojovacích vrstev, aby přesně lokalizovala objekty a klasifikovala je do relevantních kategorií.

**Klíčová vylepšení** zahrnují implementaci mozaikové augmentace dat (viz Obrázek 5.9), která během tréninku simuluje variabilnější scény a vede tak k robustnějšímu modelu. Součástí jsou také techniky Bag of Freebies, které zlepšují tréninkovou strategii bez ovlivnění inferenčních nákladů. Příkladem je ztrátová funkce Complete Intersection over Union (*angl. CIoU*) [37], která zvyšuje přesnost lokalizace obalových obdélníků. Další významnou součástí jsou metody Bag of Specials, které mohou zvýšit inferenční náklady, ale výrazně zlepšují přesnost modelu, například prostřednictvím použití aktivace Mish [21]. Tato aktivace je integrována do architektury CSPDarknet53, což zvyšuje přesnost při udržení rychlosti inferenčního procesu. CSP architektura je dále upravena pro lepší zpracování různorodých scénářů díky efektivnější kombinaci rysů z různých vrstev pomocí PAN, což vylepšuje lokalizaci a klasifikaci objektů.

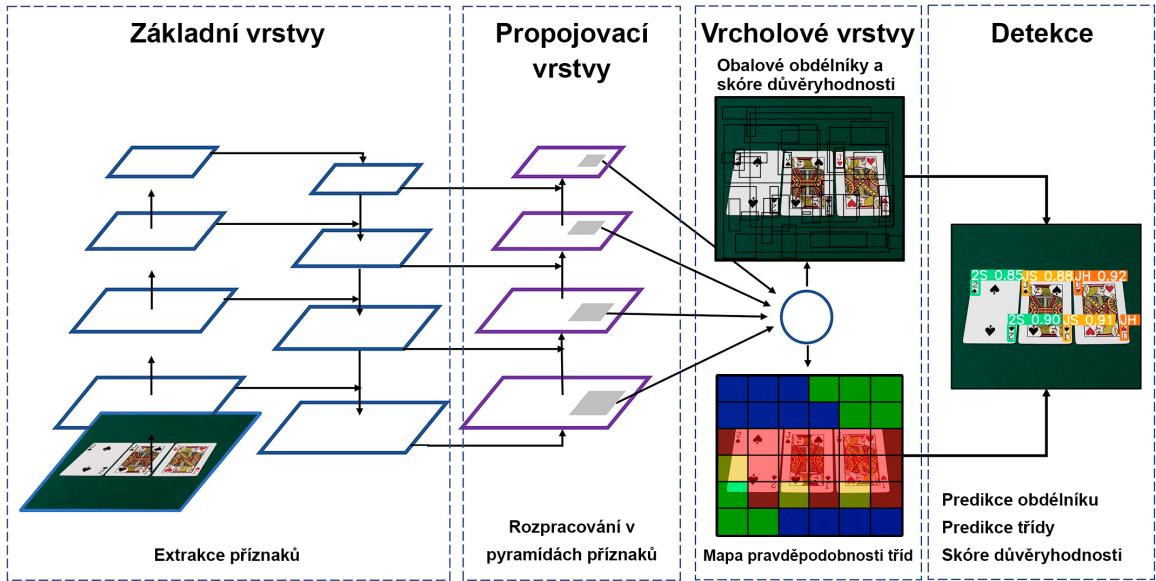
	YOLOv4	YOLOv5
Základní vrstva	CSPDarknet53	Nový CSP-Darknet53
Propojovací vrstva	SPP, PANet	SPPF, Nový CSP-PAN
Vrcholová vrstva	YOLOv3	YOLOv3
Ztrátové funkce	CIoU	BCE, CIoU

Tabulka 3.1: Porovnání struktur YOLOv4 a YOLOv5.

## YOLOv5

Model přináší zásadní změnu tím, že opouští aplikační rámec Darknet [25] a přechází na PyTorch [24]. Tento krok využívá výhody PyTorch, jako jsou lepší integrace s nástroji pro umělou inteligenci, zjednodušení vývojového procesu a udržitelnost kódu, a umožňuje lepší využití grafických procesorů, což vede k rychlejšímu tréninku a inferenci.

**Architektonické změny** zahrnují implementaci struktury CSP-PAN (vylepšená PAN), která zjednoduší model, přičemž zachovává efektivitu v kombinaci rysů na různých úrovních. SPP bylo nahrazeno vrstvou SPPF (F pro rychlosť), což zvyšuje receptivní pole a lepší odděluje nejdůležitější rysy od základní vrstvy. Dále model integruje vrstvu  $6 \times 6$  Conv2d, která efektivně redukuje rozměry obrazu a zvyšuje hloubku kanálů. Tato úprava vede k redukci počtu vrstev a parametrů, čímž zrychluje zpracování obrazu. Tabulka 3.1 uvádí porovnání klíčových komponent struktur YOLOv4 a YOLOv5. Proces zpracování dat je zobrazen na Obrázku 3.10.



Obrázek 3.10: Architektura YOLOv5. Obrázek byl adaptován z [3].

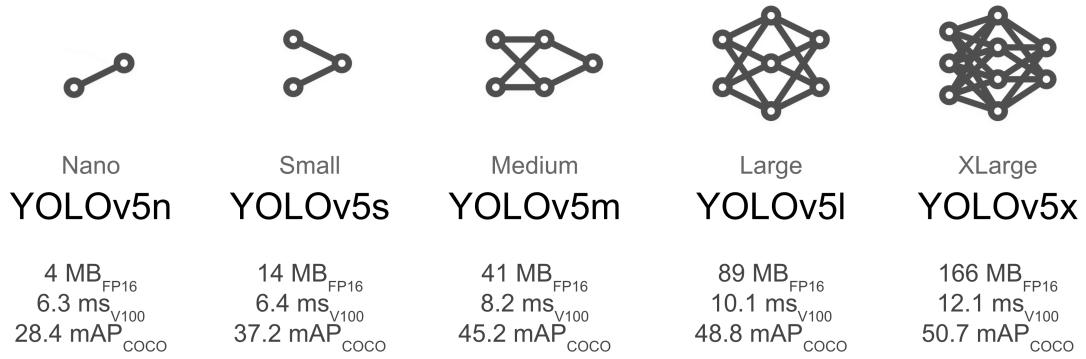
**Ztrátová funkce** je vypočítána jako součet tří specifických komponent ztrát:

$$Ztráta = \lambda_1 Z_{cls} + \lambda_2 Z_{obj} + \lambda_3 Z_{loc}, \quad (3.2)$$

kde  $\lambda_1, \lambda_2, \lambda_3$  jsou hyperparametry reprezentující váhové koeficienty pro jednotlivé složky ztráty, které jsou nastaveny na hodnoty 0.05, 0.7 a 0.3.

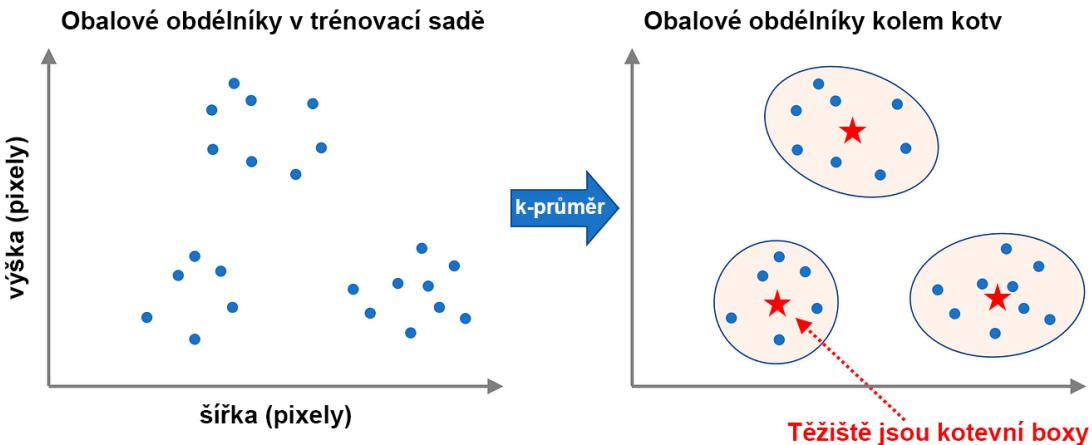
První komponenta  $Z_{cls}$ , Binary Cross-Entropy (*angl. BCE*) ztráta, se používá k měření chyby v klasifikaci, přičemž každá třída je posuzována jako přítomná nebo nepřítomná. Druhá komponenta  $Z_{obj}$ , rovněž BCE ztráta, vyhodnocuje chybu v detekci přítomnosti objektu v určité buňce mřížky. Třetí komponenta  $Z_{loc}$ , CIoU ztráta, se soustředí na přesnost lokalizace objektů v mřížce. Tato kombinace složek ztrát umožňuje efektivně a přesně identifikovat a klasifikovat objekty a současně minimalizovat chyby v lokalizaci.

**Konfigurace velikostí modelu** nabízí pět variant velikostí, které se liší ve hloubce sítě a šířce mapy příznaků. Tato rozmanitost umožňuje uživatelům vybrat optimální rovnováhu mezi rychlostí a přesností detekce podle specifických potřeb jejich aplikací. Charakteristiky těchto modelů jsou zobrazeny na Obrázku 3.11.



Obrázek 3.11: Pět základních variant velikostí modelu YOLOv5. Obrázek byl převzat z [34].

**Optimalizace kotvících boxů metodou k-průměrů** umožňuje efektivněji detekovat objekty různých tvarů a rozměrů. Tento proces, zobrazený na Obrázku 3.12, ukazuje, jak metoda k-průměrů kategorizuje ohraničení do skupin podle šířky a výšky v pixelech. Hvězdy na obrázku reprezentují těžiště, která definují optimální kotvící boxy sloužící jako referenční body pro detekci objektů. Díky tomu se model lépe adaptuje na specifické charakteristiky objektů v datové sadě, což vede ke zvýšení celkové přesnosti detekce.



Obrázek 3.12: Optimalizace kotvících boxů pomocí metody k-průměrů v YOLOv5. Obrázek byl adaptován z [8].

## Základy trénování modelu YOLOv5

Trénink modelu YOLOv5 je rozsáhlý proces, který se skládá z několika důležitých kroků. V následujících odstavcích jsou popsány klíčové etapy potřebné k úspěšnému vývoji funkčního modelu.

**Příprava a organizace dat** představují první a nejdůležitější krok. Data je třeba strukturovat do formátu, který obsahuje obrázky společně s příslušnými anotacemi. Tyto anotace musí určovat polohu objektů pomocí souřadnic obalových obdélníků a specifikovat třídu, ke které daný objekt náleží.

**Velikost modelu** je pečlivě vybírána na základě dostupných výpočetních zdrojů a specifických požadavků na rychlosť a přesnost detekce. Zatímco větší modely obvykle nabízí lepší přesnost, jejich efektivní využití vyžaduje výkonné zařízení.

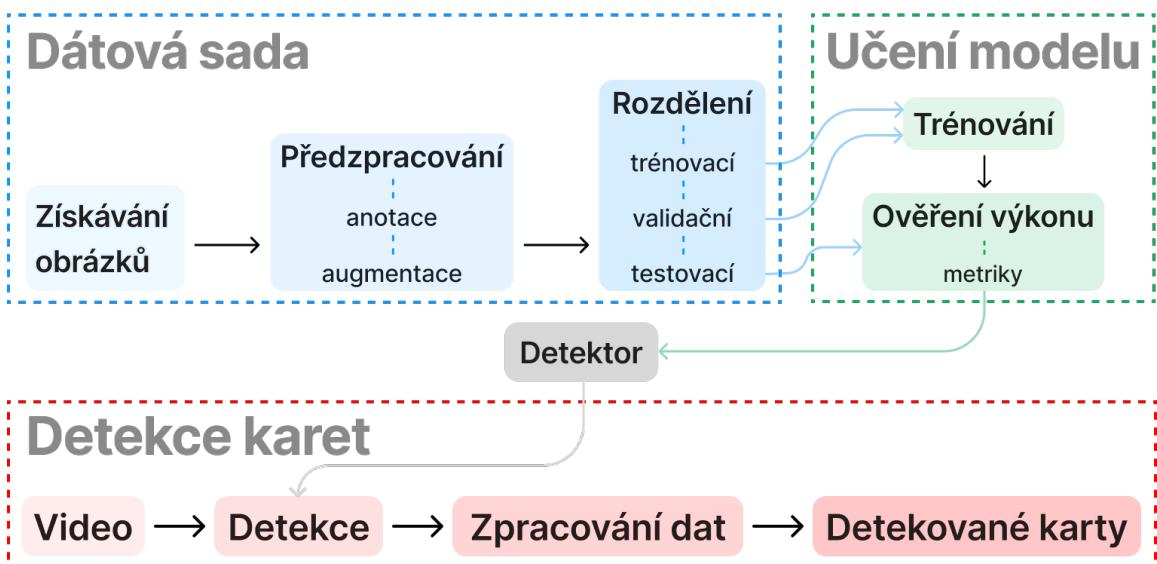
**Spuštění tréninku** začíná konfigurací parametrů, jako je velikost obrázku, velikost dávky a počet epoch. Správné nastavení těchto parametrů je nezbytné pro efektivní trénink. Trénink probíhá na grafickém procesoru pro dosažení maximálního výkonu. Během něj se model učí rozpoznávat a lokalizovat objekty na základě anotovaných dat a upravuje své vnitřní váhy na základě zpětné vazby ztrátové funkce.

**Validace a testování** jsou klíčové pro ověření generalizační schopnosti modelu na neznámá data. Tento krok zajišťuje, že model dokáže přesně detekovat a klasifikovat objekty v různých situacích. Nezávislá testovací sada dat slouží jako důležitý nástroj pro posouzení finální výkonnosti modelu před jeho nasazením v praxi.

# Kapitola 4

## Návrh řešení

V této kapitole představují koncepcii návrhu řešení pro detekci, klasifikaci a evidenci hracích karet, který jsem rozdělil do tří vzájemně propojených částí, ilustrovaných na Obrázku 4.1.



Obrázek 4.1: Proces řešení pro detekci, klasifikaci a evidenci pokerových karet. Začíná sběrem dat, následuje trénování modelu a poté aplikace modelu na videa z pokerových turnajů a zpracování získaných dat.

V první fázi jsem vytvořil rozsáhlou datovou sadu, která sloužila jako základ pro následné trénování a testování detektoru. Tato datová sada obsahuje obrázky karet, jak reálných, tak synteticky generovaných. Vzhledem k rozsahu a hloubce zpracování této fáze jsem se rozhodl věnovat jí samostatnou Kapitolu 5.

Ve druhé fázi jsem použil algoritmus YOLO pro trénování detektoru, který je schopen s vysokou přesností rozpoznávat a klasifikovat karty. Tento model jsem testoval na záznamech z pokerových turnajů, abych ověřil jeho přesnost a připravenost na reálné nasazení.

Ve třetí fázi jsem aplikoval natrénovaný model na videozáznamy pokerových turnajů a na základě klasifikovaných karet jsem vytvořil algoritmus pro evidenci rozehraných karet.

## 4.1 Koncepce a cíle systému

Rád bych začal tím, že povím o účelu systému, na kterém jsem pracoval. Cílem mé práce bylo vyvinout systém pro rozpoznávání hracích karet, které jsou aktuálně viditelné jak na stole, tak v rukou hráčů. Systém měl být schopen nejenom určit přesnou polohu každé karty, ale také identifikovat, o jaký typ karty se jedná, její barvu a hodnotu. Po úspěšném rozpoznání karet měl systém tyto informace zpracovat a uložit. To zahrnovalo evidenci všech karet, které se objevily na stole, a také těch, které měli hráči v rukou, což poskytuje užitečné údaje pro analýzu průběhu hry.

### Kamerový systém

Můj systém je organizován tak, že přijímá data ve formě videozáznamů. Tyto videozáznamy jsou získávány pomocí kamer instalovaných nad hracími stoly a zabudovaných do hran stolů každého účastníka hry.



Obrázek 4.2: Pohled z kamery shora na pokerový stůl, kde krupiér rozkládá karty. Zdroj je uveden v Tabulce 5.1.

Kamera umístěná nad stolem je instalována tak, aby mohla zachytit činnosti krupiéra při rozdávání karet, stejně jako oblast rozložení karet a malý prostor kolem ní. Pohled z této kamery je znázorněn na Obrázku 4.2.

Kamery zabudované do hran stolů jsou nasměrovány na ruce hráčů, kteří drží karty. Hráči obvykle pouze krátce pohlédnou na své karty a nedrží je otevřené po celou dobu hry. Tyto kamery jsou umístěny tak, aby hráči nemohli nahlédnout na své karty, aniž by se dostali do zorného pole kamery. Pohled z kamery je znázorněn na Obrázku 4.3.

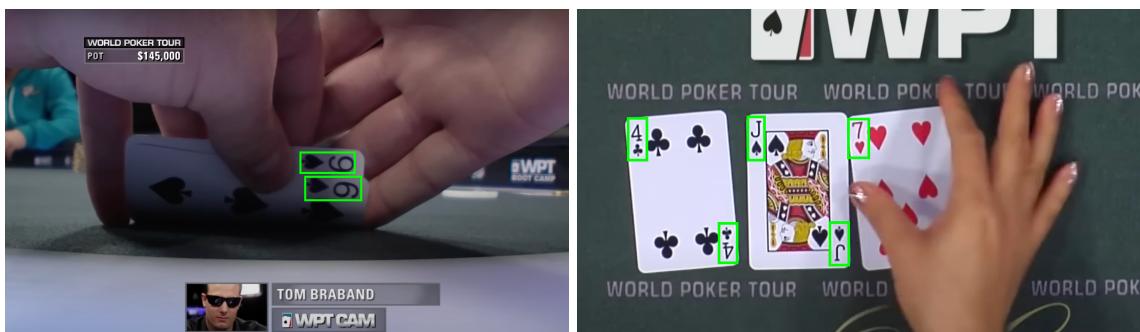


Obrázek 4.3: Pohled z kamery zabudované do hran stolu, kde hráč se dívá na své karty. Zdroj je uveden v Tabulce 5.1.

### Identifikace karet podle rohů

Zajímavou vlastností detekce karet je, že pro úspěšné rozpoznání postačuje identifikace pouze jednoho rohu karty, který obsahuje nejdůležitější informaci. Na klasické kartě jsou takové informativní rohy dva, umístěné diagonálně od sebe.

Tento přístup je zvláště užitečný pro detekci karet, které drží hráči, jak demonstruje Obrázek 4.4 vlevo. Pokusy o rozpoznání celé karty nejsou praktické, neboť hráči mají obtíže ukázat kameře celou kartu bez zkreslení. V praxi hráči často odhalují pouze část karty, aby zjistili její hodnotu a barvu, přičemž zbytek karty může být zakryt další kartou a viditelné zůstávají pouze dva rohy různých karet.



Obrázek 4.4: Detekce karet pomocí identifikace jejich rohů. Zeleným obdélníkem je vyznačena oblast obsahující veškeré informace o kartě. Zdroje jsou uvedeny v Tabulce 5.1.

I když je teoreticky možné rozpoznat celou kartu na stole, v praxi to není nutné, neboť pro získání potřebných informací postačuje analyzovat pouze jeden z rohů, jak demonstruje Obrázek 4.4 vpravo. Když krupír rozdává karty, zakrývá je částečně rukama nebo jinou kartou, což trvá přibližně dvě sekundy, a v takové situaci je analýza celé karty nepraktická. Z uvedených důvodů jsem trénoval model přímo na identifikaci rohů karet, jelikož tento přístup umožňuje dosáhnout efektivnějšího a praktičtějšího rozpoznávání karet v reálných podmínkách pokerové hry. Tento přístup však přináší další problém: pokud je karta na obrázku zobrazena celá, detektor ji identifikuje dvakrát. Řešení tohoto problému je podrobně popsáno v následující sekci.

## Evidence a zpracování informací o herních kartách

Po analýze každého snímku algoritmus nejprve detekuje a klasifikuje rohy karet. Tyto informace jsou následně předány k dalšímu zpracování. Systém musí zpracovávat tyto údaje s ohledem na specifika jednotlivých herních scénářů.

**V kontextu karet držených hráči** není rozhodující jejich přesná poloha, ale to, co detektor klasifikuje. Podle pravidel pokeru může každý hráč držet maximálně dvě karty. Pro každý vstupní videozáznam systém předpokládá, že obsahuje záběry rukou právě jednoho hráče, což umožňuje systému přiřazovat karty ke hráčům podle pořadí, v jakém jsou jednotlivá videa nahrávána. I když se karty objevují v různých časech a ukazují po jedné, systém spojuje každou kartu s příslušným hráčem. Algoritmus pro přiřazení karet hráčům je uveden ve Výpisu 4.1.

---

```
pro každý záznam
    inicializuj seznam karet pro jednoho hráče
    pro každý snímek
        detekuj karty
        pro každou kartu
            pokud hráč tuto kartu dříve neměl
                přiřaď kartu hráči
            konec podmínky
        konec cyklu
    konec cyklu
konec cyklu
```

---

Výpis 4.1: Algoritmus pro automatické přiřazení karet hráčům podle videozáznamů.

**Karty na stole** hrají klíčovou roli během celé hry i v jejím závěru. Dle pravidel může být na stole až pět karet, přičemž hra může skončit dříve, než jsou všechny karty rozloženy. Systém zvládá situaci, kdy na stole nejsou žádné karty, nebo když hra skončí s méně než pěti kartami.



Obrázek 4.5: Příklad snímku, kde je nutné použít oblast zájmu. Karty ležící mimo tuto oblast nejsou rozpoznávány, jelikož jsou považovány za nepotřebné. Zdroj je uveden v Tabulce 5.1.

Na stole se mohou nacházet i další karty, které neobsahují žádné relevantní informace a neměly by být detektorem zpracovány. Příklad je uveden na Obrázku 4.5. Jedná se o hráčské karty, které by měl systém přiřadit konkrétním hráčům ještě předtím, než se dostanou do záběru kamery snímající stůl. To znamená, že tyto karty už v systému existují a nebudou dále zpracovávány. Však existují situace, kdy kamera nemusí kartu hráče zachytit, například kvůli špatnému úhlu záběru, nebo pokud se hráč rozhodne své karty vůbec neprohlížet. Aby bylo možné tento problém řešit, bylo nutné implementovat parametr definující oblast zájmu, tedy specifickou oblast, kde mohou být karty rozpoznány.

**Deduplikace detekcí.** Ve hře nemohou být současně dvě identické karty, a proto bylo třeba zpracovat informace z rohů karet následujícím způsobem: pokud jsou na jednom snímku objeveny dva rohy s identickou informací, znamená to, že na snímku je pouze jedna karta. Pokud je objeven pouze jeden roh, jedná se o jednu kartu. Tedy v případě Obrázku 4.6, kde jsou zobrazeny devět rohů a pouze pět z nich obsahuje unikátní informaci, systém zpracuje informace o pěti kartách.

**Zpracování chybné detekce.** Absolutní bezchybnost modelu je v praxi téměř nedosažitelná. Chyby v detekci se často objevují, pokud je roh karty rozmazený nebo špatně viditelný. To bývá v případě, že se karta pohybuje, nebo je částečně zakryta. Pro řešení tohoto problému byl vyvinut algoritmus minimalizující chyby v konečném výčtu karet. Tento algoritmus má za úkol filtrovat detekce karet na videu. Detekce, kde byla karta identifikována pouze omezený počet případů, jsou z finálního výčtu karet vyrazeny. Limity pro toto filtrování byly stanoveny na základě pozorování, že chybné detekce se na videu neobjevily ve více než 20 snímcích pro kartu na stole a 5 snímcích pro kartu hráče.

**Systém ukládá informace** z každého snímku, včetně čísla snímku, klasifikace a polohy rohů karet, a pravděpodobnosti jejich rozpoznání. Po skončení hry se uloží všechny shromážděné údaje do JSON souboru, což umožňuje snadnou analýzu dat. Na výpisu 4.2 je demonstrován příklad uložení těchto informací o snímku z Obrázku 4.6. Na základě informací z tohoto souboru systém poté vytváří textový dokument, který obsahuje přehled karet držených hráči nebo karet na stole.



Obrázek 4.6: Fragment videa, jehož JSON soubor je uveden ve Výpisu 4.2. Zdroj je uveden v Tabulce 5.1.

---

```
{  
    "frame_id": 275,  
    "timestamp": 9.14,  
    "cards_detected_one_corner": [  
        "4D"  
    ],  
    "cards_detected_two_corners": [  
        "AC",  
        "KC",  
        "5H",  
        "2C"  
    ],  
    "cards_info": [  
        {  
            "card_id": 37,  
            "name": "AC",  
            "x_coord": 361.53,  
            "y_coord": 427.79,  
            "confidence": 0.91  
        },  
        ...  
        {  
            "card_id": 45,  
            "name": "KC",  
            "x_coord": 513.7,  
            "y_coord": 425.11,  
            "confidence": 0.86  
        }  
    ]  
}
```

---

Výpis 4.2: Ukázka obsahuje část informace o konkrétním snímku (viz Obrázek 4.6) z videa v datovém formátu JSON. Tyto informace zahrnují data o daném snímku a detaile o detekované kartě.

Po analýze celého souboru JSON, jehož část je prezentována ve Výpisu 4.2, systém vytvořil textový dokument, který zobrazuje karty v pořadí, v jakém byly rozloženy na stole:

Karty na stole: AC, KC, 5H, 2C, 4D.

## 4.2 Trénování a testování modelu

Trénink modelu byl důležitou fází, protože nesprávný výběr algoritmu by mohl výrazně snížit účinnost modelu, pokud by neodpovídal konkrétním požadavkům úkolu. Z tohoto důvodu bylo důležité zvolit vhodnou architekturu a vyvinout metodiku pro testování detektoru. Tento proces zahrnoval určení metrik pro hodnocení efektivity modelu a provádění experimentů pro nalezení optimálního řešení.

## Vhodná architektura

Z architektur uvedených v Kapitole 3 jsem dal přednost YOLOv5 z několika důvodů. Tento model vyniká vysokou přesností a rychlým zpracováním v reálném čase, což je nezbytné pro okamžité rozpoznávání karet během pokerových turnajů.

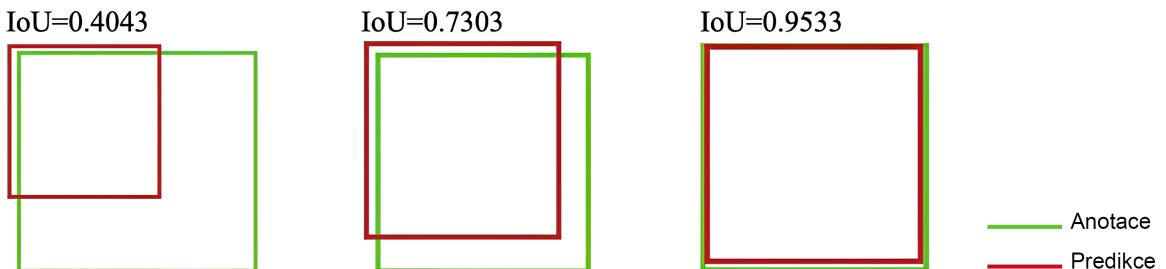
Celkově byla YOLOv5 vynikající volbou, nejen díky své architektuře, ale i díky jednoduchému a uživatelsky přívětivému designu, založenému na platformě PyTorch, kterou implementovala firma [Ultralytics](#). Tato kombinace činí z YOLOv5 ideální řešení pro specifické úlohy, jako je detekce karet v pokerových turnajích.

## Metriky hodnocení detektoru

Testování představovalo závěrečnou fázi vývoje detektoru, během níž došlo k jeho evaluaci na neznámých datech. Model jsem validoval na reprezentativní sadě testovacích dat, která obsahovala různé scénáře z pokerových turnajů, aby bylo možné ověřit jeho schopnosti v reálných podmínkách.

Pro hodnocení efektivity modelů jsem využíval průměrnou přesnost všech tříd (*angl. mAP*<sup>1</sup>) [14]. Jedná se o rozšířenou metriku, která se běžně používá v oblasti detekce objektů a poskytuje komplexní přehled o výkonu modelu. Pro výpočet mAP je třeba provést následující kroky:

**Výpočet Intersection over Union (IoU) [29].** Metrika, která měří překryv mezi předdefinovaným obalovým obdélníkem a detekovaným obalovým obdélníkem, znázorněna na Obrázku 4.7. Používá se k určení, zda je detekce správná na základě zvolené míry. Hodnota IoU se počítá jako podíl plochy průniku a plochy sjednocení obou obdélníků.



Obrázek 4.7: Míra překrytí mezi předdefinovanými a detekovanými obalovými obdélníky v závislosti na jejich pozici. Čím vyšší je hodnota IoU, tím větší je shoda mezi očekávanou a detekovanou polohou. Obrázek byl adaptován z [17].

Na základě vypočítaného IoU se pak detekce klasifikují jako správné nebo nesprávné podle toho, zda překročí stanovený práh (např. 0,5). Detekce jsou následně klasifikovány jako:

- **True Positives (TP)** jsou správně detekované karty.
- **False Positives (FP)** jsou chybně detekované karty.
- **False Negatives (FN)** jsou přehlédnuté nebo nesprávně určené karty.
- **True Negatives (TN)** jsou správně neidentifikované karty.

<sup>1</sup>mAP – *mean Average Precision* (průměrná přesnost všech tříd).

**Výpočet přesnosti a výtěžnosti.** Přesnost je metrika, která vyjadřuje poměr správně detekovaných objektů k celkovému počtu objektů detekovaných modelem. Tato hodnota ukazuje, do jaké míry model poskytuje falešně pozitivní výsledky, to znamená, kolik z detekovaných objektů je ve skutečnosti nesprávných. Na druhou stranu, výtěžnost měří poměr správně detekovaných objektů vzhledem k celkovému počtu skutečně existujících objektů. Tato metrika poukazuje na to, do jaké míry model „přehlíží“ existující objekty, to znamená, kolik reálně existujících objektů model nedetekuje. Tyto metriky lze vyjádřit následujícími rovnicemi:

$$Přesnost = \frac{TP}{TP + FP}, \quad (4.1)$$

kde  $TP$  značí počet správně detekovaných karet a  $FP$  počet chybně detekovaných karet,

$$Výtěžnost = \frac{TP}{TP + FN}, \quad (4.2)$$

kde  $FN$  značí počet přehlédnutých nebo nesprávně určených karet.

**Výpočet průměrné přesnosti (AP) a mAP.** Pro každou třídu objektů se vypočítává AP, metrika zahrnující přesnost a výtěžnost při různých prahových hodnotách detekce. Tato metrika se určuje jako plocha pod křivkou, která graficky znázorňuje vztah mezi přesností a výtěžností. Výpočet AP probíhá integrací této křivky podle vzorce

$$AP = \int_0^1 p(u) du, \quad (4.3)$$

kde  $p(u)$  je funkce přesnosti jako funkce výtěžnosti  $u$ .

Průměr AP, vypočítaný pro všechny třídy, je označován jako mAP a poskytuje celkový výkon modelu při detekci objektů z více tříd. Pro výpočet se používá vzorec

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i, \quad (4.4)$$

kde  $N$  je počet tříd, a  $AP_i$  je průměrná přesnost pro  $i$ -tou třídu.

Pro hodnocení výkonu detekčních modelů jsem použil následující metriky:

- **mAP50** zohledňuje přesnost detekce s prahem IoU 0.5, kde detekce jsou správné, pokud míra překryvu je alespoň 50 %.
- **mAP50-95** poskytuje průměrnou přesnost vypočtenou pro IoU prahy od 0.5 do 0.95 a umožňuje získat komplexnější pohled na schopnost modelu lokalizovat objekty.

## Experimenty

Pro ověření účinnosti a přesnosti navrženého detektoru hracích karet pro pokerové turnaje byla provedena řada důležitých experimentů. Tyto experimenty měly za cíl identifikovat nejfektivnější konfiguraci detektoru a prokázat jeho výkonnost při rozpoznávání a evidenci hracích karet.

**Experiment 1: Vliv datových kombinací.** Cílem prvního experimentu bylo zjistit, jak různé kombinace dat ovlivňují schopnost detekce hracích karet. Bylo vytvořeno a testováno deset detekčních modelů trénovaných na různých typech dat, včetně syntetických dat, reálných dat a jejich kombinací. Výkonnost každého modelu byla následně hodnocena na testovací sadě složené výhradně z reálných dat.

**Experiment 2: Rychlosť detekcie.** Druhý experiment se zaměřil na analýzu různých konfigurací modelů detektoru a jejich vliv na rychlosť a přesnosť detekcie. Cílem bylo nalézt optimální model pro použití v reálném čase během pokerových turnajů. Modely byly trénovány na stejně datové sadě a následně byly vyhodnoceny na základě jejich schopnosti rychle a přesně detektovat hrací karty.

**Experiment 3: Přepis a evidence herních karet.** Třetí experiment se zaměřil na ověření správnosti evidence hracích karet pomocí simulace pokerového turnaje. Celý proces byl navržen tak, aby se ověřila funkčnost a správnost systému v reálných podmínkách.

Experimenty prokázaly funkčnost a přesnosť navrženého systému pro rozpoznávání hracích karet v podmínkách pokerového turnaje. Pomocí těchto experimentů byl identifikován nejfektivnější model vhodný pro praktické nasazení. Podrobnější výsledky a analýzy experimentů jsou uvedeny v Kapitole 7.

# Kapitola 5

## Dátová sada

Příprava datové sady byla důležitá pro účinnost modelu, jelikož jeho výkon závisí na kvalitě dat. Bylo nutné shromáždit tisíce obrázků hracích karet a provést jejich důkladnou anotaci, což vyžadovalo značné množství času a pozornosti. Kvalita datové sady má přímý vliv na schopnost modelu se učit a následně fungovat v reálných aplikacích.

Tuto kapitolu zahájím přehledem veřejně dostupné datové sady obsahující obrázky pokerových karet. Z omezeného výběru jsem zvolil sadu z projektu Playing Cards Computer Vision [2], která je dostupná na platformě Roboflow [10]. Tato datová sada obsahuje 24 233 synteticky generovaných obrázků pokerových karet umístěných na různorodých pozadích. Karty jsou prezentovány ve formě vějířů, což simuluje držení karet v ruce. Příklady těchto ilustrací naleznete na Obrázku 5.1. Data jsou anotována tak, že obalové obdélníky pokrývají pouze rohy karet, což, jak jsem již zmínil v sekci 4.1, považuji za účinnou metodu pro identifikaci a klasifikaci pokerových karet.



Obrázek 5.1: Příklady synteticky generovaných obrázků pokerových karet, získaných z datové sady na platformě Roboflow.

Během experimentů, které jsou podrobně popsány v Kapitole 7, se ukázalo, že tato datová sada není pro detekční úkoly na pokerových turnajích dostačně účinná. V důsledku toho bylo rozhodnuto rozšířit tuto sadu o syntetická a reálná data. Proces vývoje a rozšiřování datové sady je detailně popsán v této kapitole.

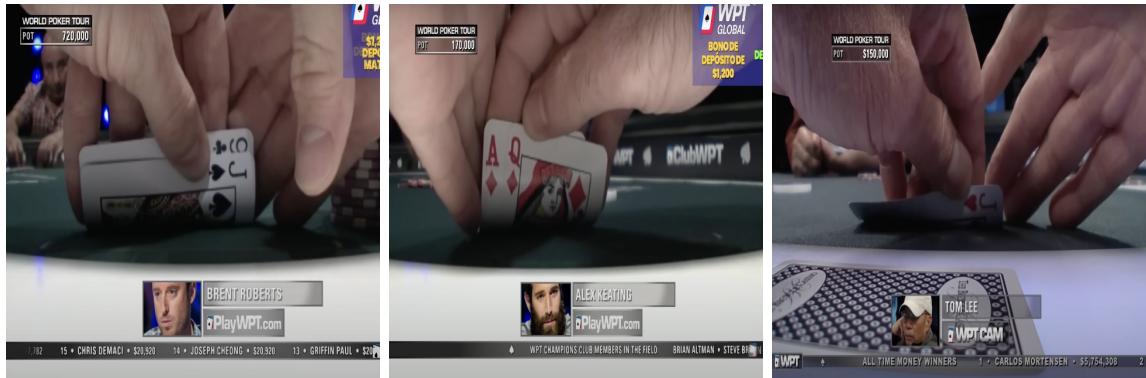
## 5.1 Zdroj dat

Efektivní trénování modelu pro rozpoznávání pokerových karet vyžaduje pečlivě zvolený přístup ke sběru dat. Během shromažďování obrázků jsem se zaměřil na optimalizaci generalizačních schopností modelu, sledujíc scénáře popsané v sekci 2.2, které zdůrazňují hlavní výzvy detekce karet v pokeru. Datová sada obsahuje různé typy pokerových karet (od dvojek po esa) ve všech barvách, zobrazené na různých pozadích a z různých perspektiv, čímž umožňuje modelu efektivně rozpoznávat karty v jakýchkoli herních situacích. Zdroje dat zahrnují videozáznamy z pokerových turnajů a vlastní fotografie.



Obrázek 5.2: Příklady obrázků pokerových karet, umístěných na herním stole, získaných z videozáznamů pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1.

**Reálná data** pocházející z autentických herních situací jsou nezbytná pro trénink detekčního modelu pokerových karet. Z tohoto důvodu byly obrázky shromázděny přímo z prostředí, ve kterém bude detektor používán, a to z pokerových turnajů, konkrétně z jejich videozáznamů.



Obrázek 5.3: Příklady obrázků pokerových karet, držených v rukou hráčů, získaných z videozáznamů pokerových turnajů. Zdroje jsou uvedeny v Tabulce 5.1.

Byla provedena detailní analýza několika hodin videonahrávek turnajů s cílem zachytit situace, ve kterých docházelo k různým manipulacím s hracími kartami. Bylo nutné vybrat pouze ty momenty, které zachytily kamery umístěné buď v integrovaných modulech hracích

stolů, nebo kamery snímající stoly shora. Z každého videa byly vybrány pouze unikátní snímky, aby byla datová sada co nejrozmanitější.

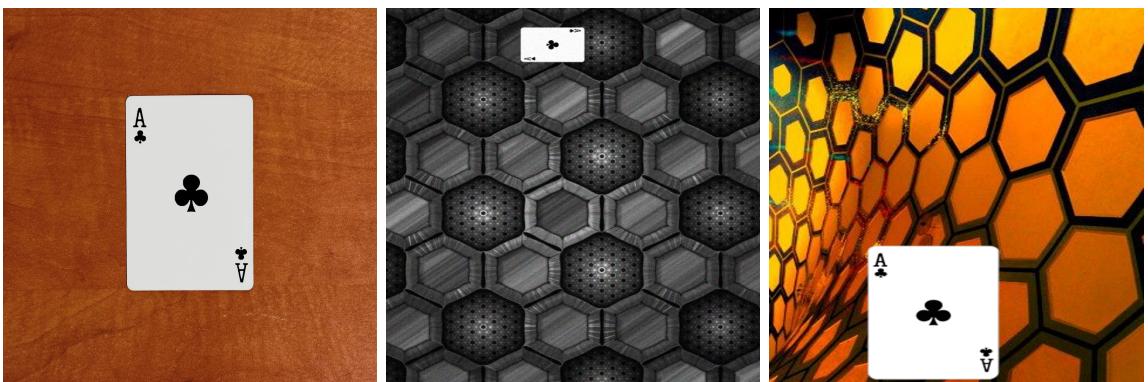
Vybrané snímky prezentují karty v širokém spektru podmínek – od ideálně čitelných až po rozmazané či částečně zakryté. Z deseti videozáznamů s celkovou délkou šest a půl hodiny bylo získáno 562 obrázků. Příklady těchto obrázků je možné vidět na Obrázku 5.2 a Obrázku 5.3. Detailní přehled původu všech reálných obrázků je poskytnut v Tabulce 5.1.

# ob.	# an.	Délka	Publikované	Název videa	Zdroj
109	445	00:31:28	24.10.2022	Freddy Deeb is Looking ...	<a href="#">YouTube</a>
103	865	00:33:50	18.10.2022	\$1,587,382 To FIRST ...	<a href="#">YouTube</a>
95	590	00:24:01	29.12.2021	\$2,151,072 at Legends ...	<a href="#">YouTube</a>
75	413	00:23:34	03.11.2022	Final Showdown for ...	<a href="#">YouTube</a>
72	463	00:34:18	19.10.2022	High Stakes \$4,000,000 ...	<a href="#">YouTube</a>
52	265	02:26:32	03.04.2024	\$4,960,000 Prize Pool: ...	<a href="#">YouTube</a>
41	186	00:27:14	24.11.2023	\$2,062,015 at WPT ...	<a href="#">YouTube</a>
8	45	00:17:37	30.11.2021	THREE-WAY ALL IN ...	<a href="#">YouTube</a>
4	25	00:20:44	22.06.2022	From FANBOY To ...	<a href="#">YouTube</a>
3	24	00:31:56	08.05.2023	\$5,133,700 Prize Pool ...	<a href="#">YouTube</a>

Tabulka 5.1: Přehled videí použitých pro extrakci obrázků. Tabulka obsahuje informace o počtu obrázků, počtu anotovaných objektů, délce záznamů, datu publikace a zdroji.

Sběr reálných dat je časově náročný proces, a proto bylo rozhodnuto využít syntetická data k rozšíření a diverzifikaci datové sady.

**Syntetická data** definovaná jako uměle generované obrázky se používají k rozšíření tréninkových souborů a zároveň šetří čas při jejich vytváření. Jednou z hlavních nevýhod této metody je občasná nedostatečná realističnost, která může vést k tomu, že modely se učí na artefaktech nebo vzorech, které ve skutečnosti neexistují. Abych tento problém minimalizoval, zahrnul jsem do tréninkového procesu také reálná data a augmentovaná reálná data.



Obrázek 5.4: První fotografie je originální snímek, zatímco napravo od něj jsou umístěny obrázky, které byly generovány na jeho základě.

Vzhledem k tomu, že datová sada již obsahovala syntetické obrázky karet, které vypadaly jako držené v ruce, bylo nutné vytvořit obrázky karet, které by imitovaly jejich umístění na stole.

V prvním kroku tvorby syntetické datové sady byla každá hrací karta vyfotografována na tmavém pozadí, čímž vzniklo celkem 52 snímků. Na Obrázku 5.4 vlevo lze vidět jeden z těchto snímků. Fotografování probíhalo zrcadlovkou Nikon D7100 s univerzálním objektivem, což zajistilo vysokou kvalitu pořízených fotografií. Poté byly jednotlivé karty odděleny od pozadí s využitím grafického editoru Adobe Photoshop a uloženy ve formátu PNG. Následně byly karty ručně anotovány, jak je podrobněji popsáno v sekci 5.2.



Obrázek 5.5: Příklady synteticky generovaných obrázků pokerových karet, které jsem vytvořil pro rozšíření datové sady.

Bylo vybráno několik desítek pozadí, většinou jednobarevných, imitujících povrch pokerového stolu. Dalším krokem bylo vytvoření programu pro generování obrázků, který využíval vybraná pozadí, karty a jejich anotace. Program prováděl různé transformace obrázků, včetně změny velikosti, přidání rozmazání, úpravy kontrastu, jasu a šumu. Tyto augmentační techniky jsou podrobně popsány v sekci 5.4. Nakonec bylo vygenerováno 6968 variabilních obrázků, jejichž příklady lze nalézt na Obrázku 5.5.

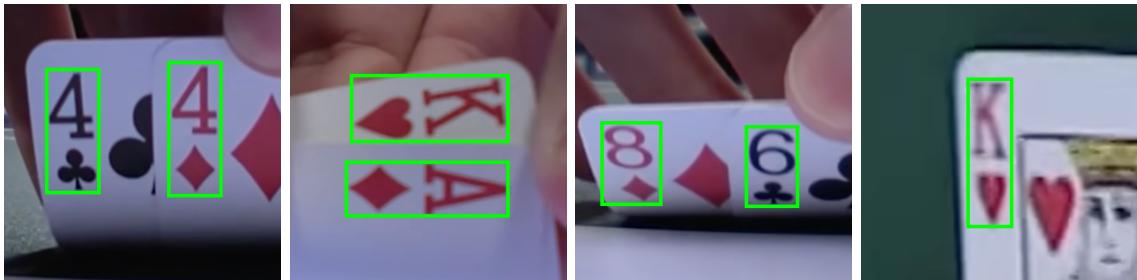
Další fáze přípravy datové sady spočívala v anotaci každého obrázku a je popsána v následující sekci.

## 5.2 Vývoj a implementace anotačních algoritmů

Anotace obrázků představuje základní proces přidávání popisných informací k obrázkům, čímž se výrazně usnadňuje a zefektivňuje jejich analýza počítačem. Tento proces začíná označováním obrázků buď ručně člověkem, nebo automaticky pomocí programu. Po anotaci následuje revize, která zajišťuje přesnost dat, což je klíčové pro spolehlivé učení modelu. Anotovaná data je rovněž nezbytná pro hodnocení modelu po jeho tréninku. Tento proces je často iterativní s cílem dosáhnout optimální kvality dat.

Pro anotaci obrázků jsem použil obalové obdélníky (viz Obrázek 5.6), což bylo zejména vhodné při anotaci rohů pokerových karet. Proces anotace byl rozdělen do tří etap.

**Ruční anotace.** Nejprve bylo nutné definovat anotační třídy, přičemž každá třída reprezentovala specifickou kartu. Názvy tříd vycházely z kombinace nominální hodnoty a barvy karty, například 10S pro desítku pik a KH pro krále srdcí. Celkem bylo definováno 52 tříd.



Obrázek 5.6: Ukázka přesného zachycení hodnot a barev karet pomocí obalových obdélníků. Zdroje jsou uvedeny v Tabulce 5.1.

Každý obrázek jsem ručně zkoumal, identifikoval rohy karet a přiřadil je do příslušných tříd. Pro přesné umístění obalových obdélníků kolem rohů karet byla využita platforma Roboflow [10]. Bylo nezbytné zajistit, aby obalový obdélník těsně obepínal hodnotu a barvu karty, i když je karta otočená. Požadovaná přesnost umístění obdélníků je demonstrována na Obrázku 5.6.

Tato fáze si vyžádala značné množství času. Například anotace rohů na 100 obrázcích s celkem 350 kartami (přibližně 600 rohů) trvala dvě hodiny. Vzhledem k náročnosti manuálního procesu byly zahájeny snahy o jeho částečnou automatizaci.

**Poluautomatizovaná anotace.** Druhá fáze zahrnovala trénování modelu na základě anotovaných dat s cílem automatizovat anotaci nových obrazů. Tento krok vedl k výraznému zrychlení anotačního procesu, avšak v některých případech se vyskytly problémy s nesprávnou identifikací a klasifikací rohů karet. Pro řešení těchto nedostatků byly vyvinuty dvě pomocné aplikace:

- **Filtracní program** sloužil k filtrování obrázků podle tříd a byl používán k ověření, zda všechny vyfiltrované obrázky správně odpovídají popisným informacím uvedeným u každého obrázku.
- **Vizualizační nástroj** umožňoval zobrazit obalové obdélníky na obrazech, což usnadňovalo verifikaci jejich správného umístění.

Celkem bylo takto anotováno 462 obrázků obsahujících přibližně 2750 fragmentů rohů karet.

**Automatizovaná anotace.** Třetí etapa byla spojena s tvorbou syntetických obrázků. Program pro generování syntetických obrázků, který byl vyvinut v sekci 5.1 a dále rozšířen v sekci 5.4, umožňoval rozsáhlou manipulaci s kartami – náhodné umisťování, změnu měřítka a rotaci. Z tohoto důvodu jsem navrhl algoritmus, který automaticky upravoval anotace v závislosti na změnách polohy a orientace karet v syntetických scénářích. Pseudokód tohoto algoritmu je uveden ve Výpisu 5.1.

Díky tomuto automatizovanému přístupu bylo možné generovat data ve velkém množství, čímž se výrazně zvětšil tréninkový soubor a v konečném důsledku to vedlo ke zlepšení výkonu modelu, jak prokázaly experimenty v Kapitole 7.

---

```
pro každou kombinaci karet a pozadí
    získej rozměry pozadí
    urči náhodnou pozici pro kartu na pozadí
    inicializuj seznam pro nové koordináty štítků
    pro každý štítok
        vypočti nové středové koordináty štítku na pozadí
        přepočti šířku a výšku štítku podle nové velikosti karty
        přidej nové koordináty do seznamu
    ulož obrázek a nové koordináty štítků
konec cyklu
```

---

Výpis 5.1: Algoritmus pro automatickou aktualizaci dat v anotačních souborech na základě změn polohy karet.

### 5.3 Následné zpracování

Shromážděná data pro trénování modelu YOLOv5 je nutné uložit v požadovaném formátu. Obrázky jsou ve standardních formátech, jako je JPEG nebo PNG, s rozměry 640x640 pixelů, což optimalizuje rychlosť trénování modelu.

Každý obrázek v datové sadě je spárován s jedním textovým souborem (.TXT) obsahujícím údaje, které definují polohu a klasifikaci objektů. Anotační formát YOLOv5, známý jako **YOLO Darknet TXT** formát, používá zápis ve struktuře:

```
<třída> <X středu> <Y středu> <šířka> <výška>
```

kde **<třída>** označuje identifikátor třídy objektu, **<X středu>** a **<Y středu>** jsou souřadnice středu objektu na obrázku a **<šířka>** a **<výška>** udávají rozměry objektu v pixelech. Všechny souřadnice a rozměry jsou normalizovány vzhledem k šířce a výšce obrázku, což zajišťuje konzistence dat bez ohledu na jejich původní rozměry. Každý řádek v anotačním souboru reprezentuje jeden objekt.

### 5.4 Augmentace dat

Augmentace dat představuje důležitou techniku v trénování detektorů, která spočívá v modifikaci stávajících tréninkových dat za účelem generování nových a rozmanitých obrázků. Cílem je rozšířit datovou sadu a zvýšit její diverzitu, čímž se zlepšuje schopnost modelu generalizovat na neviděná data.

#### Předtréninková augmentace

Pro rozšíření datové sady jsem použil augmentační techniky na již existující reálné obrázky. Tyto techniky jsem aplikoval v různých pořadích, například nejprve rotace a pak změna jasu, aby data byla ještě variabilnější.

**Augmentace na úrovni pixelů.** Tato kategorie zahrnuje metody, které manipulují s barvou, intenzitou a dalšími vlastnostmi pixelů obrázku bez změny jeho původního prostorového uspořádání. Na Obrázku 5.7 a Obrázku 5.8 jsou demonstrovány různé techniky, které jsou dále detailně popsány.



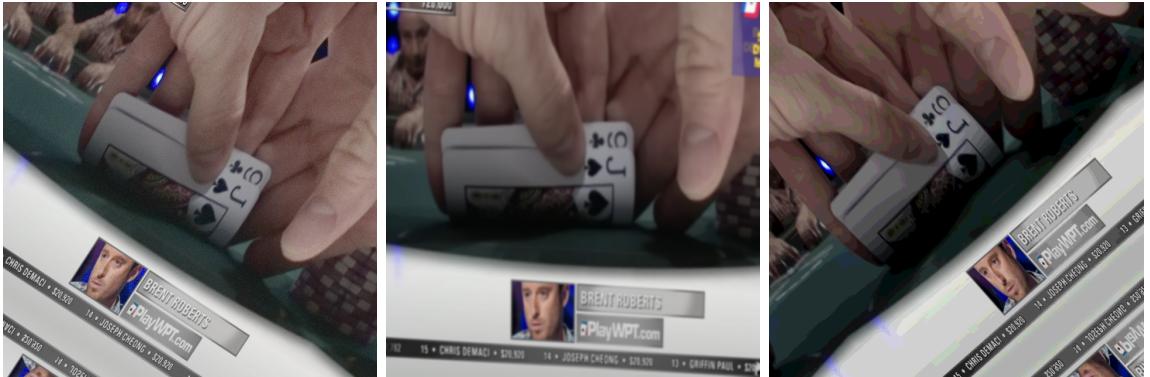
Obrázek 5.7: Příklady různých augmentačních technik aplikovaných na stejný obrázek (původní obrázek je zobrazen na Obrázku 5.2 vpravo). Zleva doprava: obrázek po aplikaci rotace a úpravy odstínu a sytosti barev, obrázek po deformaci a posunutí, obrázek po aplikaci černobílého filtru a zmenšení. Zdroj je uveden v Tabulce 5.1.

- **Změna jasu:** Tato technika upravuje celkovou světlost obrázku, čímž umožňuje modelu lépe se adaptovat na různé světlenné podmínky.
- **Úprava odstínu a sytosti barev:** Tato technika mění barevné tóny a sytost na obrázku, což pomáhá modelu lépe generalizovat znalosti na různé barvy.
- **Aplikace černobílého filtru:** Tato metoda odstraňuje barvy z obrázku, čímž zlepšuje obecné schopnosti modelu, protože se učí rozpoznávat objekty bez závislosti na jejich barvě.
- **Přidání zrnitosti:** Tato technika simuluje efekt vyššího ISO na fotografie, čímž umožňuje modelu lépe fungovat i za zhoršených světlenných podmínek.
- **Rozmazání:** Tato technika napodobuje efekt pohybu kamery nebo špatného zaostření, čímž napomáhá modelu učit se rozpoznávat rozmazané objekty.

**Augmentace na úrovni prostorových transformací.** Tato kategorie zahrnuje metody, které manipulují s umístěním a orientací objektů na obrázku, čímž se simulují různé pohledové úhly a situace, se kterými se detektor může setkat v reálném světě. Na Obrázku 5.7 a Obrázku 5.8 jsou demonstrovány níže uvedené techniky:

- **Rotace:** Obrázek je otočen o náhodný úhel, čímž se model naučí rozpoznávat objekty z různých orientací.
- **Posunutí:** Obrázek je posunut horizontálně i vertikálně, čímž se simuluje změna polohy objektů v záběru.
- **Deformace:** Nepravidelné změny tvaru objektů na obrázku napodobují různé deformace a zkreslení, které se vyskytují v reálném světě.
- **Změna perspektivy:** Tato technika upravuje úhel pohledu na objekty, čímž se simuluje pohled z různých výšek nebo úhlů. To zlepšuje schopnost modelu interpretovat objekty z různých perspektiv.

- **Změna velikosti:** Zmenšení nebo zvětšení obrázků umožňuje modelu naučit se detektovat objekty různých velikostí a na různých vzdálenostech.



Obrázek 5.8: Příklady různých augmentačních technik aplikovaných na stejný obrázek (původní obrázek je zobrazen na Obrázku 5.3 vlevo). Zleva doprava: obrázek po aplikaci rotace a přidání zrnitosti, obrázek po aplikaci rozostření a změny perspektivy, obrázek po aplikaci posterizace a deformace. Zdroj je uveden v Tabulce 5.1.

Kromě reálných obrázků jsem také do programu pro generování syntetických obrázků (ze sekce 5.1) integroval některé z uvedených augmentačních technik, což významně přispělo k rozšíření datové sady.

Výsledkem těchto kroků je, že datová sada nyní obsahuje 24 233 obrázků z otevřených zdrojů a 14 734 pečlivě vybraných, anotovaných a augmentovaných obrázků, na kterých jsem pracoval. Podrobný popis obsahu dat je uveden v Tabulce 5.2.

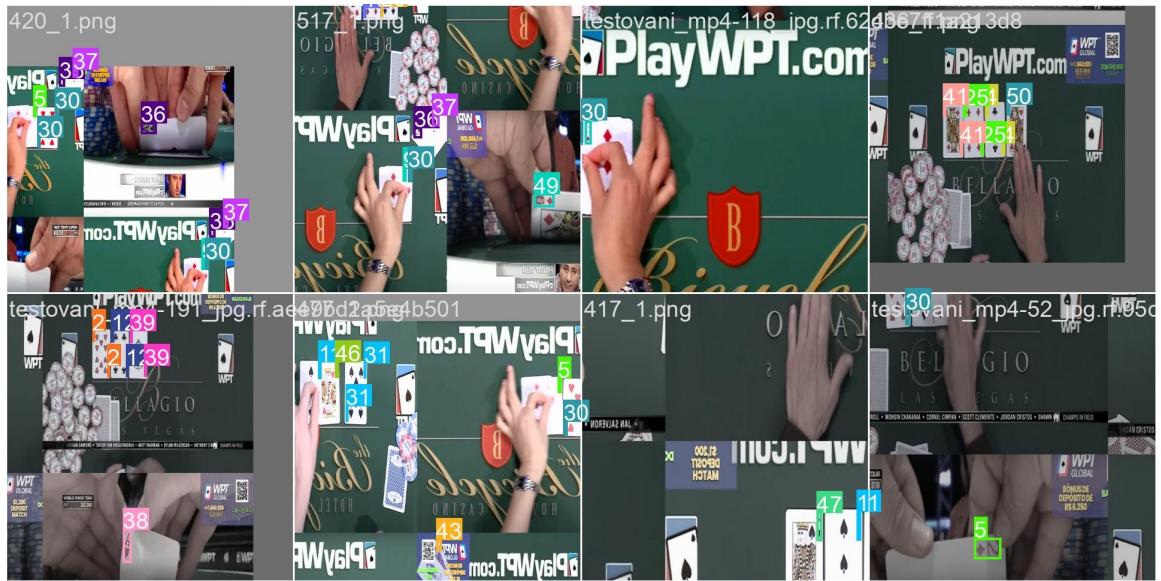
Vlastní shromážděná data	
Reálná (bez úprav)	562
Reálná (rozšířená augmentací)	7208
Syntetická	6964
Data z externího zdroje	
Syntetická	24233
$\Sigma$	<b>38967</b>

Tabulka 5.2: Přehled datové sady.

## Augmentace v YOLOv5

Model YOLOv5 implementuje augmentaci dat přímo během tréninkového procesu. Tím se zajišťuje, že každý tréninkový krok je unikátní.

Nejčastěji používanou technikou je mozaiková augmentace, která kombinuje čtyři různé obrázky do jednoho. Tato technika rozšiřuje zkušenosť modelu vůči objektům různých velikostí a kontextů, což je důležité pro detekci menších objektů. Příklad mozaikové augmentace a dalších augmentací používaných v YOLOv5 lze vidět na Obrázku 5.9.



Obrázek 5.9: Ukázka augmentace obrázků při tréninku detekčního modelu YOLOv5. Na obrázku je zobrazeno osm augmentovaných snímků, které zahrnují mozaikovou augmentaci, modifikace barev a další techniky.

## 5.5 Rozdělení dat

Posledním krokem v procesu přípravy datové sady bylo rozdělení dat do samostatné tréninkové, validační a testovací sady.

**Tréninková sada** sloužila k trénování modelu. Jedná se o největší část datové sady, která umožnila modelu naučit se a přizpůsobit se různorodým situacím. Tréninková sada tvořila 85 % celkového objemu dat, čímž poskytla modelu dostatečnou variabilitu pro učení.

**Validační sada** sloužila k optimalizaci parametrů modelu a k předběžnému hodnocení jeho výkonu. Jejím hlavním účelem bylo identifikovat případy, kdy se model příliš přizpůsobil trénovacím datům, což by mohlo naznačovat přeúčení. Tato sada představovala 15 % z celkového objemu dat.

**Testovací sada** byla určena pro konečné hodnocení modelu. Tato data nebyla použita během tréninku ani validace a sloužila k ověření, jak dobře model funguje na zcela neznámých datech. Pro tento účel bylo vyhrazeno 350 unikátních obrázků, které pocházely přímo z pokerových turnajů a nebyly nijak augmentovány.

Pro rozdělení dat jsem použil vlastní program, který náhodně přiřazuje obrázky do testovací a validační sady na základě předem definovaných poměrů. Náhodný výběr zajišťuje reprezentativnost sad, což je zásadní pro vyvážené trénování detektoru.

V další fázi proběhla série experimentů s cílem nalézt optimální poměr syntetických a reálných dat v tréninkové sadě. Podrobnější informace o těchto experimentech jsou uvedeny v Kapitole 7.

# Kapitola 6

## Implementace

Tato kapitola se zabývá technickým popisem vývoje systému pro rozpoznávání hracích karet, od vytvoření datové sady až po jeho testování v simulovaných podmírkách pokerových turnajů.

Systém byl implementován v programovacím jazyce [Python](#) verze 3.10, který byl vybrán pro jeho robustní funkční možnosti a rozsáhlou podporu knihoven v oblasti strojového učení. Mezi klíčové technologie použité v této bakalářské práci patří objektová detekce za použití architektury [YOLOv5](#), integrace této modelu do systému s využitím knihovny [PyTorch](#) a zpracování obrazu prostřednictvím [OpenCV](#). Další použité technologie a jejich specifické funkce jsou popsány v následujících sekcích.



Obrázek 6.1: Ukázka anotace obrázku na platformě Roboflow.

**Sběr dat.** Pro trénování a testování modelu byla vytvořena datová sada obsahující reálné obrázky z videonahrávek pokerových turnajů dostupných na platformě [YouTube](#). Tato data byla zásadní pro trénink modelu za reálných podmínek, se kterými se model setká v praxi. Datová sada byla dále rozšířena o syntetické obrázky získané z platformy [Universe Roboflow](#), čímž se dosáhlo větší diverzity a rozsahu trénovacích příkladů.

**Anotace a augmentace dat.** Anotace reálných obrázků probíhala pomocí platformy [Workspace Roboflow](#), která umožňuje přesné umístění obalových obdélníků. Uživatelské rozhraní této platformy během anotace obrázku je zobrazeno na Obrázku 6.1. Augmentace těchto obrázků byla provedena pomocí nástroje [YOLO Data Augmentation](#), který s využitím knihovny Albumentations [5] efektivně upravuje obalové obdélníky v souladu se změnami v obrázcích. Generování syntetických dat probíhalo pomocí algoritmů aplikujících náhodné efekty za použití knihoven [PIL](#), [NumPy](#) a [Random](#), zatímco pro vizualizaci obalových obdélníků byla využita knihovna [OpenCV](#).

**Vytváření modelu.** Vývoj modelu zahrnoval průběžné upravování poměrů syntetických a reálných dat v datové sadě. K tomuto účelu byly použity knihovny [Shutil](#) a [Random](#), které zajišťovaly potřebné dělení dat a organizaci adresářů.

Jako základní algoritmus pro detektor byl zvolen [YOLOv5](#), a to pro jeho vysokou výkonost a přesnost v detekci objektů v reálném čase. Trénování modelů probíhalo na výkonnému školním serveru Sophie GPU vybaveném grafickými kartami Nvidia RTX A5000, což vedlo k výraznému zkrácení doby tréninku. Pro optimalizaci procesu a izolaci projektu bylo využito [virtuálního prostředí Python](#). K zajištění neustálého běhu tréninkových procesů bez přerušení, i po odhlášení z terminálu, byl používán příkaz [nohup](#), což umožnilo bezproblémové pokračování v trénování na pozadí.



Obrázek 6.2: Vytvořený zobrazovací nástroj pro zobrazení detekovaných karet na herním stole a v rukách hráčů. Snímek z videa demonstruje ikony rozpoznaných karet, které jsou viditelné v levém horním rohu. Na snímku je také uvedeno jeho číslo. Zdroj je uveden v Tabulce 5.1.

**Použití modelu a zpracování výsledků.** Systém využívá knihovnu [Argparse](#) pro parsování cesty k videosouboru zadанé z příkazové řádky. Model [YOLOv5](#), načítaný pomocí knihovny [PyTorch](#), analyzuje každý snímek videa a detekuje hrací karty s jistotou vyšší než

80 %. Současně s tímto systém vytváří z analyzovaných snímků nové video, do kterého přidává ikony detekovaných karet pomocí knihovny [OpenCV](#). Výsledky detekce, včetně typu karty, souřadnic a jistoty rozpoznání, jsou uloženy do [JSON](#) souboru. Závěrem systém generuje textový soubor se seznamem detekovaných karet a vytváří upravené video s vloženými ikonami těchto karet. Ukázkový snímek z náhodného videa, kde jsou tyto ikony vidět, je zobrazen na Obrázku [6.2](#).

# Kapitola 7

## Výsledky experimentů

Tato kapitola se zaměřuje na sérii experimentů, jejichž cílem je identifikovat nejfektivnější detektor hracích karet pro použití během pokerové hry a ověřit funkčnost systému v simulovaném prostředí pokerového turnaje.

### 7.1 Experiment 1: Vliv datových kombinací

Hlavním cílem prvního experimentu bylo zjistit, jaký vliv mají různé kombinace dat na detekci hracích karet. Bylo vytvořeno a otestováno deset detekčních modelů, které byly trénovány s využitím syntetických dat z externích zdrojů, vlastních syntetických dat, reálných dat a augmentovaných reálných dat. Za základ modelů sloužily předtrénované váhy z modelu **YOLOv5m** na datové sadě **COCO**. Trénování probíhalo s dávkou o velikosti 128. K vyhodnocení všech modelů byla použita trénovací sada obsahující 350 unikátních obrázků získaných z videozáZNAMŮ pokerových turnajů.

#	Počet dat					$\Sigma$	Váhy	Výsledky	
	Syn.	Ext.	Syn.	Real.	Aug. re.			mAP50	mAP50-95
1	20598	0	0	0	0	20598	COCO	0.394	0.084
2	0	5919	0	0	0	5919	COCO	0.688	0.413
3	5919	5919	0	0	0	11838	COCO	0.804	0.457
4	0	0	180	0	0	180	COCO	0.032	0.022
5	0	0	0	6307	6307	6307	COCO	0.559	0.400
6	20598	5919	0	0	0	26517	COCO	0.853	0.504
7	20598	5919	0	6307	32824	COCO	<b>0.987</b>	<u>0.724</u>	
8	0	0	0	6307	6307	Tab. 7.1 ř. 6	0.952	0.669	
9	20598	5919	0	3154	29671		COCO	<u>0.986</u>	<b>0.735</b>
10	20598	5919	180	0	26697		COCO	0.969	0.703

Tabulka 7.1: Výsledková tabulka s porovnáním deseti detekčních modelů z Experimentu 1. Nejlepší výsledky jsou zvýrazněny tučně a podtrženy. Kurzívou je zvýrazněn první model, který slouží jako výchozí bod a byl převzat z externího zdroje. Označení: Syn. Ext. (syntetická data z externího zdroje), Syn. (syntetická data), Real. (reálná data), Aug. re. (augmentovaná reálná data).

**Analýza výsledků.** V Tabulce 7.1 jsou prezentovány podrobné výsledky pro všechny konfigurace modelů, včetně hlavních hodnotících metrik mAP50 a mAP50-95 (viz sekci 4.2).

Modely trénované výhradně na syntetických nebo reálných datech dosáhly horších výsledků než modely, které kombinovaly oba typy dat. Referenční model využívající pouze syntetická data z externího zdroje (detektor č. 1) vykázal extrémně nízké hodnoty mAP50 a mAP50-95, což ukazuje jeho omezenou schopnost detekce hracích karet ve srovnání s ostatními modely. Model trénovaný s vlastními syntetickými daty (detektor č. 2) dosáhl lepších, ale stále neuspokojivých výsledků. Naproti tomu detektor č. 6 kombinující všechna dostupná syntetická data (vlastní i externí) dosáhl významně lepších výsledků.

Detektor č. 4, který používal pouze malé množství reálných dat, vykázal velmi nízké hodnoty mAP. Oproti tomu modely trénované pouze s augmentovanými reálnými daty dosáhly lepších výsledků, což zdůrazňuje důležitost dostatečného množství a rozmanitosti tréninkových dat.

Nejlepší výsledky byly získány kombinací syntetických a reálných dat, která vedla k výraznému zlepšení generalizačních schopností modelu. Detektor č. 10, i při použití malého množství reálných dat, prezentoval vynikající výsledky, což svědčí o účinnosti augmentace integrované přímo do modelu YOLO. Nicméně pro dosažení nejlepšího výsledku bylo nezbytné zvýšit množství reálných dat. Detektory č. 7 a 9, které využívaly největší objem syntetických a augmentovaných reálných dat, dosáhly nejvyšších hodnot mAP50 a mAP50-95, což potvrzuje, že velké množství reálných dat přidané k syntetickým zlepšuje detekční výkonnost v různých scénářích.

Přestože detektor č. 8, trénovaný výhradně na augmentovaných reálných datech s využitím předtrénovaných vah z detektoru č. 6 (trénovaného na všech dostupných syntetických datech), dosáhl uspokojivých výsledků, stále zaostával za detektory č. 7 a 9. Tato odlišnost naznačuje, že přímá integrace všech dostupných dat do tréninkového procesu vede k lepším výsledkům než postupný trénink s využitím předtrénovaných vah. Jednofázový trénink umožňuje komplexnější učení modelu, s čímž souvisí i vyšší celková přesnost detekce.

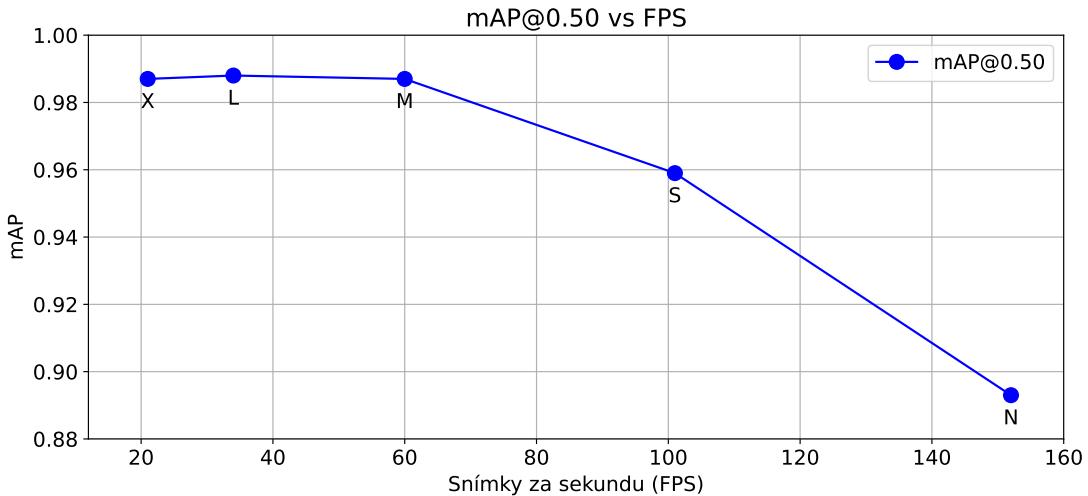
## 7.2 Experiment 2: Rychlosť detekcie

Druhý experiment byl zaměřen na analýzu vlivu různých konfigurací velikosti modelů na rychlosť a přesnosť detekcie. Hlavním cílem bylo nalezení nejoptimálnějšího modelu pro použití v podmírkách pokerových turnajů v reálném čase.

Experiment probíhal na grafické kartě NVIDIA GeForce RTX 2060. Všechny modely byly trénovány na stejné datové sadě jako model č. 7 (Tab. 7.1 ř. 7) z prvního experimentu. Pro trénování však byly použity různé velikosti modelů YOLOv5 s přetrénovanými váhami na datové sadě MS COCO, které byly zmíněny v sekci 3.5.

Předtrénované váhy	mAP50	mAP50-95	Čas/snímek (ms)	Snímky/s
YOLOv5n-COCO	0.893	0.625	6.6	152
YOLOv5s-COCO	0.959	0.684	9.9	101
YOLOv5m-COCO	<u>0.987</u>	<b>0.724</b>	16.7	60
YOLOv5l-COCO	<b>0.988</b>	0.715	29.4	34
YOLOv5x-COCO	<u>0.987</u>	0.711	48.1	21

Tabulka 7.2: Výsledková tabulka s porovnáním rychlosti detekcie modelů YOLOv5 v Experimentu 2.



Obrázek 7.1: Graf porovnávající frekvenci zpracování snímků (*angl. FPS*) a mAP50 modelů YOLOv5 v rámci Experimentu 2. Označení: X (YOLOv5x), L (YOLOv5l), M (YOLOv5m), S (YOLOv5s), N (YOLOv5n).

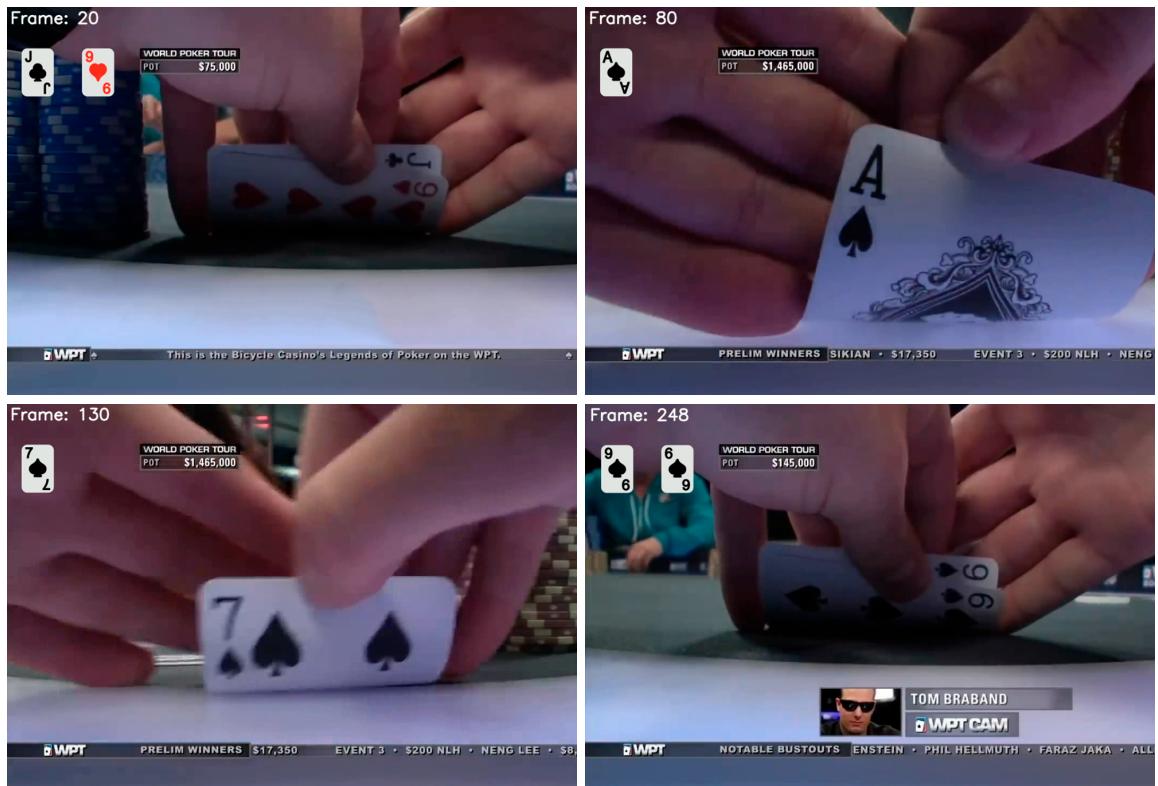
**Analýza výsledků.** V Tabulce 7.2 jsou prezentovány podrobné výsledky pro všechny konfigurace modelů, včetně metrik mAP50 a mAP50-95, a také čas zpracování jednoho snímku, což je zásadní pro aplikace v reálném čase.

Na základě Obrázku 7.1 lze pozorovat, že model YOLOv5m vykazuje nejoptimálnější poměr mezi vysokou přesností a dostatečnou rychlostí detekce, což je zásadní pro použití v dynamickém prostředí pokerových turnajů. I když modely YOLOv5l a YOLOv5x vykazovaly stejnou nebo mírně vyšší hodnotu mAP50, jejich delší doba zpracování jednoho snímku je činí méně vhodnými. Naopak model YOLOv5n má nejvyšší rychlosť zpracování, ale jeho hodnoty mAP jsou výrazně nižší, což může vést k problémům při detekci jemnějších detailů hracích karet.

Výsledky jasně ukazují, že model YOLOv5m představuje optimální řešení pro rychlé a přesné rozpoznávání hracích karet v reálném čase. Díky jeho schopnosti efektivně zpracovávat snímky s vynikajícími hodnotami mAP50-95 je tento model vhodný pro dynamické podmínky pokerových turnajů, kde je nezbytná rychlá adaptace na změny a spolehlivá detekce.

### 7.3 Experiment 3: Evidence herních karet

Cílem třetího experimentu bylo ověřit správnost evidování karet prostřednictvím simulovaného zápasu v rámci pokerového turnaje. Za tímto účelem bylo nutné spojit několik nezávislých videozáznamů do jednoho logicky uceleného videa, které chronologicky zobrazuje rozdávání karet krupiérem, jejich prohlížení hráči a následné vykládání krupiérem karet na stůl až do konce hry. Pro simulaci různých typů kamerových záznamů bylo video rozděleno do dvou segmentů: jeden pro karty držené hráči (viz Obrázek 7.2) a druhý pro karty na stole (viz Obrázek 7.3).



Obrázek 7.2: Výseky z videa zachycující karty různých hráčů. Zdroje jsou uvedeny v Tabulce 5.1.



Obrázek 7.3: Výsek z videa, zachycující rozkládání karet krupiérem na herním stole. Zdroj je uveden v Tabulce 5.1.

**Analýza výsledků.** Systém po analýze videí s hráči generoval textový soubor, který přesně specifikoval karty držené každým hráčem, čímž ukázalo funkčnost a správnost systému. V tomto případě soubor dat obsahoval:

Hráč 1: 9H, JC.

Hráč 2: AS, 7S.

Hráč 3: 6S, 9S.

což odpovídalo očekávaným výsledkům. Video, na kterém krupiér rozkládal karty na stůl, rovněž ukázalo správné výsledky:

Karty na stole: AC, 6S, 2S, 9D, QD.

Ačkoli počet karet na videu se zdál být vyšší než počet karet uvedených v textovém souboru, výsledky byly považovány za správné, protože systém nezapočítal karty mimo zaměřenou oblast (viz sekci 4.1).

Testování tohoto přístupu bylo provedeno na vzorku pěti her, což umožnilo systému prokázat své schopnosti v různých scénářích. V každém testovacím případě systém spolehlivě zaznamenal informace o kartách, čímž demonstroval svou schopnost přesně fungovat v podmínkách napodobujících reálné herní situace. Tento experiment potvrdil jeho validitu a praktickou aplikovatelnost s konečnou přesností dosahující 100 %. Tyto výsledky zdůrazňují potenciál systému jako spolehlivého nástroje pro automatické zaznamenávání karet v dynamických a složitých podmínkách pokerových her.

# Kapitola 8

## Záver

Cílem této bakalářské práce bylo vyvinout systém schopný detektovat, identifikovat a evidovat herní karty během pokerových turnajů s využitím moderních technik zpracování obrazu. Tento záměr byl úspěšně splněn prostřednictvím implementace konvolučních neuronových sítí, které umožnily přesné rozpoznávání karet.

V rámci práce byly postupně řešeny úkoly definované ve formálním zadání. Systém pro detekci karet byl navržen, implementován a následně otestován na speciálně připravené a upravené rozsáhlé datové sadě, která byla augmentována a adaptována k zajištění robustnosti a účinnosti detekce. Iterativní proces vylepšování detektoru umožnil dosáhnout přesnosti rozpoznávání karet na jednotlivých snímcích na úrovni 98,7 %. Evidence karet po analýze videozáznamů pokerových zápasů ukázala vždy správné údaje. Tyto výsledky potvrzují, že navržený systém má potenciál pro využití i v praxi.

Během práce na projektu jsem získal cenné zkušenosti s pokročilými technologiemi zpracování obrazu a hlubokého učení, které jsou stále více využívány v průmyslových aplikacích. Tato zkušenosť mi umožnila nejen rozšířit mé technické dovednosti, ale také lépe porozumět možnostem a omezením moderního počítačového vidění.

Přestože výsledky práce byly pozitivní, existuje několik možností pro jeho další rozvoj a vylepšení. Hlavním cílem budoucího vývoje by mělo být rozšíření systému o zpracování živého vysílání ze všech kamer. Velkým přínosem by také bylo začlenění mechanismu pro automatické učení detektoru, který by umožnil systému adaptovat se na nově nastalé podmínky a zlepšovat jeho schopnost rozpoznávat nové vzory hracích karet bez nutnosti manuálního zásahu.

Další možností vylepšení je integrace systému do platformy rozšířené reality. To by umožnilo na základě rozpoznaných karet analyzovat a vizualizovat herní statistiky a pravděpodobnosti v reálném čase. Hráči by tak získali hlubší vhled do průběhu hry a vývoje herních situací. Tento nový přístup by byl obzvláště přínosný pro začínající pokerové hráče, kteří se snaží lépe porozumět strategiím a procesu rozhodování v této komplexní hře.

Závěrem lze říci, že práce na tomto projektu byla pro mě velmi obohacující a nejen přinesla řadu praktických i teoretických poznatků, ale také mě inspirovala k dalšímu prohloubování mých znalostí v oblasti počítačového vidění a umělé inteligence. Věřím, že výsledky této práce mohou poskytnout praktická řešení pro reálné aplikace a mohou sloužit jako základ pro budoucí vývoj, který by mohl zásadně zlepšit vnímání a analýzu pokerových her.

# Literatura

- [1] AGARAP, A. F. *Deep Learning using Rectified Linear Units (ReLU)*. 2019.
- [2] AUGMENTED STARTUPS. *Playing Cards Dataset* [online]. Roboflow, 2023. Dostupné z: <https://universe.roboflow.com/augmented-startups/playing-cards-ow27d>. [cit. 2024-04-02].
- [3] AZAM, M. A.; SAMPieri, C.; IOPPI, A.; AFRICANO, S.; VALLIN, A. et al. Deep Learning Applied to White Light and Narrow Band Imaging Videolaryngoscopy: Toward Real-Time Laryngeal Cancer Detection. *The Laryngoscope*, 2022, sv. 132, č. 9, s. 1798–1806.
- [4] BOCHKOVSKIY, A.; WANG, C.-Y. a LIAO, H.-Y. M. *YOLOv4: Optimal Speed and Accuracy of Object Detection*. 2020.
- [5] BUSLAEV, A.; IGLOVIKOV, V. I.; KHVEDCHENYA, E.; PARINOV, A.; DRUZHININ, M. et al. Albumentations: Fast and Flexible Image Augmentations. *Information*, 2020, sv. 11, č. 2. ISSN 2078-2489.
- [6] CANNY, J. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986, sv. 8, č. 6, s. 679–698.
- [7] CASTILLO, M.; GOEING, B. a WESTELL, J. *Computer Vision for Card Games*. 2017. Stanford University, Department of Electrical Engineering.
- [8] CHERNYTSKA, O. *Training YOLO? Select Anchor Boxes Like This* [online]. 18. srpna 2022. Dostupné z: <https://towardsdatascience.com/training-yolo-select-anchor-boxes-like-this-3226cb8d7f0b>. [cit. 2024-04-12].
- [9] DUDA, R. O. a HART, P. E. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM*. New York, USA: Association for Computing Machinery, Leden 1972, sv. 15, č. 1, s. 11–15. ISSN 0001-0782.
- [10] DWYER, B.; NELSON, J.; HANSEN, T. et al. *Roboflow* [online]. 2024. Dostupné z: <https://roboflow.com>. [cit. 2024-03-02]. Software pro počítačové vidění.
- [11] ERHAN, D.; SZEGEDY, C.; TOSHEV, A. a ANGUELOV, D. *Scalable Object Detection using Deep Neural Networks*. 2013.
- [12] GIRSHICK, R. Fast R-CNN. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, s. 1440–1448.

- [13] HE, K.; ZHANG, X.; REN, S. a SUN, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. In: *Lecture Notes in Computer Science*. Springer International Publishing, 2014, s. 346–361. ISBN 9783319105789.
- [14] HENDERSON, P. a FERRARI, V. *End-to-end training of object class detectors for mean average precision*. 2017.
- [15] HRDLIČKA, J. *Detekce a rozpoznávání pokerových karet v digitálním obrazu* [online]. 2017. Bakalářská práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd. Dostupné z: <https://theses.cz/id/ebpv9x/>. [cit. 2024-04-15].
- [16] JHA, S.; DEY, A.; KUMAR, R. a KUMAR SOLANKI, V. A Novel Approach on Visual Question Answering by Parameter Prediction using Faster Region Based Convolutional Neural Network. *International Journal of Interactive Multimedia and Artificial Intelligence*, 2019, sv. 5, č. 5, s. 30–37. ISSN 1989-1660.
- [17] LEE, S.-H. a CHEN, H.-C. U-SSD: Improved SSD Based on U-Net Architecture for End-to-End Table Detection in Document Images. *Applied Sciences*, 2021, sv. 11, č. 23. ISSN 2076-3417.
- [18] LIU, S.; QI, L.; QIN, H.; SHI, J. a JIA, J. *Path Aggregation Network for Instance Segmentation*. 2018.
- [19] LIU, W.; ANGUELOV, D.; ERHAN, D.; SZEGEDY, C.; REED, S. et al. *SSD: Single Shot MultiBox Detector*. 2015.
- [20] MATHWORKS. *Introduction to Deep Learning: What Are Convolutional Neural Networks?* [online]. 24. března 2017. Dostupné z: <https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html>. [cit. 2024-04-12].
- [21] MISRA, D. *Mish: A Self Regularized Non-Monotonic Activation Function*. 2020.
- [22] O’SHEA, K. a NASH, R. *An Introduction to Convolutional Neural Networks*. 2015.
- [23] OTSU, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 1979, sv. 9, č. 1, s. 62–66.
- [24] PASZKE, A.; GROSS, S.; MASSA, F.; LERER, A.; BRADBURY, J. et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019.
- [25] REDMON, J. *Darknet: Open Source Neural Networks in C*. 2016.
- [26] REDMON, J.; DIVVALA, S.; GIRSHICK, R. a FARHADI, A. *You Only Look Once: Unified, Real-Time Object Detection*. 2016.
- [27] REDMON, J. a FARHADI, A. *YOLOv3: An Incremental Improvement*. 2018.
- [28] REN, S.; HE, K.; GIRSHICK, R. a SUN, J. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. 2016.
- [29] REZATOFIGHI, H.; TSOI, N.; GWAK, J.; SADEGHIAN, A.; REID, I. et al. *Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression*. 2019.

- [30] SCHMIDHUBER, J. Deep learning in neural networks: An overview. *Neural Networks*. Elsevier BV, Leden 2015, sv. 61, s. 85–117. ISSN 0893-6080.
- [31] SIMONYAN, K. a ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015.
- [32] SNYDER, D. *Playing Card Detection and Identification*. 2019.
- [33] TAHIR, A.; MUNAWAR, H. S.; AKRAM, J.; ADIL, M.; ALI, S. et al. Automatic Target Detection from Satellite Imagery Using Machine Learning. *Sensors*, 2022, sv. 22, č. 3. ISSN 1424-8220.
- [34] ULTRALYTICS. *YOLOv5: A state-of-the-art real-time object detection system* [online]. 2021. Dostupné z: <https://docs.ultralytics.com>. [cit. 2024-04-04].
- [35] WANG, C.-Y.; LIAO, H.-Y. M.; YEH, I.-H.; WU, Y.-H.; CHEN, P.-Y. et al. *CSPNet: A New Backbone that can Enhance Learning Capability of CNN*. 2019.
- [36] YAMASHITA, R.; NISHIO, M.; DO, R. K. G. a TOGASHI, K. Convolutional neural networks: an overview and application in radiology. *Insights Imaging*. Springer Science and Business Media LLC, Srpen 2018, sv. 9, č. 4, s. 611–629.
- [37] ZHENG, Z.; WANG, P.; LIU, W.; LI, J.; YE, R. et al. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, Duben 2020, sv. 34, č. 07, s. 12993–13000.

## Příloha A

# Obsah přiloženého paměťového média

<b>app/</b>	Zdrojové soubory aplikace a soubor <b>README.md</b> s pokyny ke spuštění
<b>datasets/</b>	Datové sady použité pro trénink a testování detektoru
<b>doc/</b>	Zdrojové soubory technické dokumentace
<b>utils/</b>	Skripty používané v průběhu práce
<b>video.mp4</b>	Video prezentující dosažené výsledky