



FAKULTA APLIKOVANÝCH VĚD  
ZÁPADOČESKÉ UNIVERZITY  
V PLZNI



ZÁPADOČESKÁ  
UNIVERZITA  
V PLZNI

ZÁPADOČESKÁ UNIVERZITA

FAKULTA APLIKOVANÝCH VĚD

# Vyhodnocení kvality stehů KKY/ZDO

Jan Čedík, Daniel Tauš

19.05.2024

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Data</b>	<b>2</b>
<b>3</b>	<b>Řešení</b>	<b>4</b>
3.1	Použité metody . . . . .	4
3.1.1	Matematická morfologie . . . . .	4
3.1.2	Hranové detektory, prahování a filtrace . . . . .	4
3.1.3	Náhodný generátor čísel . . . . .	5
3.1.4	Konvoluční neuronové sítě . . . . .	5
3.2	Výsledky experimentů . . . . .	5
3.2.1	Hranové detektory spolu s morfologickými operacemi . . .	5
3.2.2	Náhodný počet stehů . . . . .	7
3.2.3	Konvoluční neuronové sítě . . . . .	7
<b>4</b>	<b>Spuštění</b>	<b>8</b>
4.1	Spustitelný skript . . . . .	8
4.2	Interaktivní notebook . . . . .	8
<b>5</b>	<b>Závěr</b>	<b>9</b>

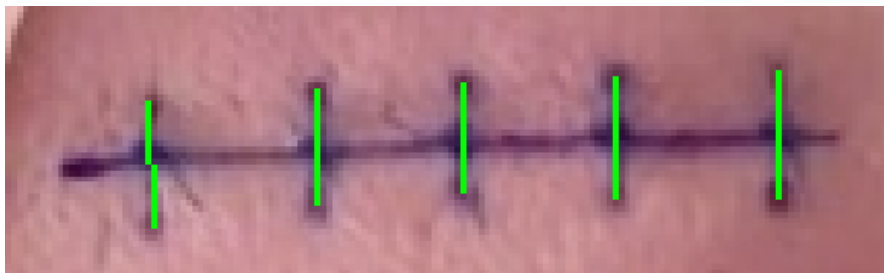
# 1 Úvod

V této semestrální práci se zaměřujeme na vývoj a testování algoritmů pro analýzu kvality stehů na prasečích končetinách, které byly použity při tréninku studentů. Celý proces zahrnuje experimenty a trénování na anotovaném datasetu, který obsahuje fotografie těchto končetin.

Veškeré kódy a podklady tohoto projektu jsou plně dostupné na naší stránce na GitHubu ([https://github.com/honzacedik/ZDO\\_2024\\_TEAM3.git](https://github.com/honzacedik/ZDO_2024_TEAM3.git)). Kódy jsou pečlivě komentovány, aby uživatelé snadno pochopili funkci každého parametru a mohli efektivně využít naše metody pro své vlastní potřeby.

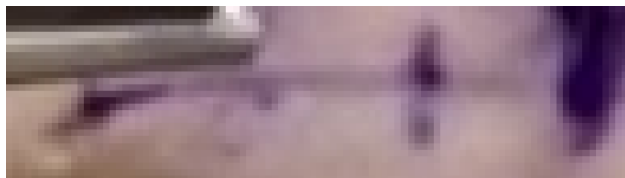
## 2 Data

Celý proces experimentů a trénování byl prováděn na datasetu, který obsahoval fotografie prasečích končetin, na nichž studenti trénovali stehy. Obrázky byly ručně anotovány v programu CVAT. Dataset obsahoval složku se 267 obrázky spolu s XML souborem, který obsahoval veškeré informace o obrázcích, jako jsou pozice bodů stehů a řezu, název, velikost a datum vytvoření. Tento pečlivě anotovaný dataset poskytoval základ pro trénování a testování algoritmů, které jsou schopny automaticky analyzovat a hodnotit kvalitu stehů na základě vizuálních dat.



Obrázek 1: Ukázka vykreslení bodů anotovaných pomocí programu CVAT.

Data, se kterými jsme pracovali, byla velmi nekonzistentní. Každý obrázek se lišil v jasu, ostrosti a celkové kvalitě. Některé obrázky byly příliš světlé, zatímco jiné byly tmavé, což komplikovalo jakoukoli jednotnou analýzu. Ostrost obrázků také výrazně kolísala, od velmi ostrých až po rozmazané snímky. Kromě těchto technických problémů se v některých obrázcích vyskytovaly rušivé objekty, jako jsou peány, texty nebo jiné nástroje, které odváděly pozornost od hlavního objektu zájmu. Dalším problémem bylo, že v několika případech nebyly na fotografiích vůbec zachyceny všechny požadované stehy.



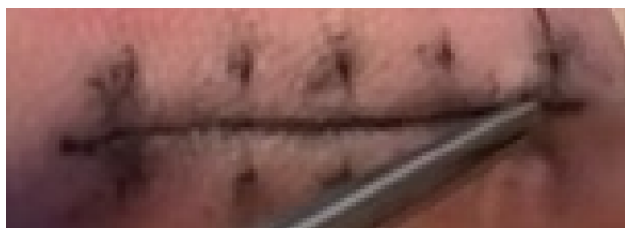
Obrázek 2: Ukázka velmi rozostřené fotky v datasetu



Obrázek 3: Ukázka fotky, kde jsou stehy mimo záběr.



Obrázek 4: Ukázka fotky, kde je část překryta štítkem.



Obrázek 5: Ukázka fotky, kde v záběru překáží nástroj.

## 3 Řešení

### 3.1 Použité metody

V této práci se zabýváme různými technikami zpracování obrazu a analýzy dat. Nejprve se věnujeme matematické morfologii, která zahrnuje operace jako dilatace a eroze pro úpravu tvarů v obraze. Dále popisujeme hranové detektory, prahování a filtraci, které jsou klíčové pro identifikaci hran a odstranění šumu. Zmiňujeme také náhodné generátory čísel. Nakonec představujeme konvoluční neuronové sítě (CNN), které jsou specializované na analýzu obrazových dat a extrakci důležitých rysů.

#### 3.1.1 Matematická morfologie

V oblasti zpracování obrazu jsou morfologické úpravy jako dilatace, eroze, otevření a uzavření základními operacemi pro analýzu a zpracování tvarů a struktur v obraze. Dilatace (*dilation*) rozšiřuje objekty v obraze, čímž vyplňuje malé mezery a propojuje blízké prvky. Naopak, eroze (*erosion*) zmenšuje objekty a odstraňuje malé výstupky, což pomáhá při odstraňování šumu. Otevření (*opening*) kombinuje erozivní a dilatační operace k odstranění malých objektů a vyhlazení hran větších objektů, aniž by se změnily jejich základní tvary. Uzavření (*closing*), které je kombinací dilatace a eroze, je užitečné pro vyplnění malých děr a spojení blízkých objektů, což pomáhá při zachování a zvýraznění hlavních struktur obrazu. Tyto morfologické operace jsou klíčové pro efektivní segmentaci, analýzu tvarů a odstranění šumu v různých aplikacích zpracování obrazu.

#### 3.1.2 Hranové detektory, prahování a filtrace

Hranové detektory jsou klíčovými nástroji ve zpracování obrazu pro identifikaci a lokalizaci hran objektů. Mezi nejpoužívanější patří Canny, Roberts a Sobel. Cannyho detektor hran je známý svou přesností a nízkou citlivostí na šum, používá více kroků včetně filtrace šumu, hledání gradientů a hysterézního prahování. Robertsův hranový detektor využívá jednoduché gradientní operátory, které jsou citlivé na diagonální hrany, ale méně účinné v přítomnosti šumu. Sobelův hranový detektor kombinuje filtraci s detekcí gradientů ve vodorovném a svislém směru, což zajišťuje robustnější výstupy oproti jednoduchým gradientním operátorům.

Prahování je technika, která převádí šedotónový obraz na binární, což je užitečné pro zvýraznění objektů před pozadím; běžné metody zahrnují globální prahování a adaptivní prahování.

Filtrace pomocí *GaussianBlur* je klíčová pro odstranění šumu a vyhlazení obrazu, čímž zlepšuje kvalitu následných zpracovatelských kroků jako je detekce hran a prahování. *GaussianBlur* používá gaussovskou funkci k rozložení hodnot pixelů, čímž vytváří jemný, rozostřený efekt, který pomáhá snížit náhodný šum a detekovat důležitější struktury v obraze.

### 3.1.3 Náhodný generátor čísel

Další použitá metoda je generátor zdánlivé náhodných čísel. Existují dva hlavní typy náhodných generátorů čísel.

- Pseudonáhodné - generují čísla na základě počátečního seedu
- Skutečně náhodné - využívají fyzikální procesy (šum, kvantové jevy)

### 3.1.4 Konvoluční neuronové sítě

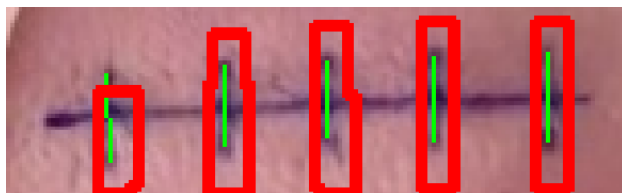
Konvoluční neuronové sítě (CNN) jsou speciální typ hlubokých neuronových sítí, které jsou navrženy pro efektivní zpracování a analýzu obrazových dat. Využívají konvoluční vrstvy, které aplikují filtry na vstupní obraz a extrahují důležité rysy, jako jsou hrany, textury a objekty. Tyto vrstvy jsou následovány poolingovými vrstvami, které snižují rozměry dat a zvyšují robustnost vůči posunům a deformacím.

## 3.2 Výsledky experimentů

### 3.2.1 Hranové detektory spolu s morfologickými operacemi

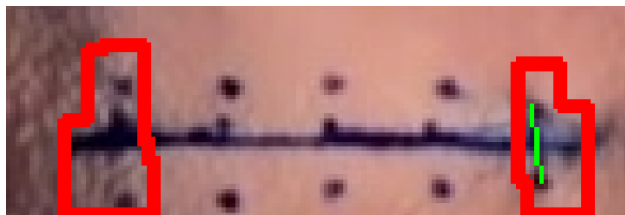
Bylo testováno několik kombinací morfologických úprav spolu s dalšími úpravami vstupního obrázku. Potýkali jsme se s několika překážkami, jako jsou nekonzistentnost obrázků, stíny, jiné objekty, špatné osvětlení, kvalita fotky, příliš tlusté náčrty fixou a špatně viditelné stehy. Tyto problémy jsme se snažili obejít právě kombinováním morfologických úprav. Nejlépe dopadla metoda `detectionMorph()`, kdy se přesnost pohybovala okolo 50%.

Na následujících obrázcích lze vidět, jakým způsobem vypadá detekce stehů pomocí metody `detectionMorph()`. Na první obrázku je bez chybně provedená detekce. Druhý obrázek demonstruje chybu, se kterou se detektor může potýkat. V tomto případě se jedná o problém s jasnými a tučnými čarami.



Obrázek 6: Detekce stehů pomocí funkce `detectionMorph()`

Další obrázky ukazují funkčnost dalších testovaných detektorů.



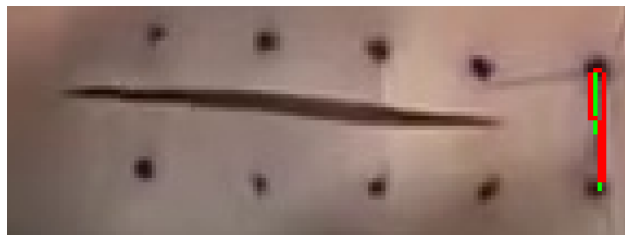
Obrázek 7: Detekce stehů pomocí funkce detectionMorph()



Obrázek 8: Detekce stehů s využitím canny detektoru



Obrázek 9: Detekce stehů s využitím canny detektoru



Obrázek 10: Detekce stehů s využitím sobel detektoru



Obrázek 11: Detekce stehů s využitím sobel detektoru

### 3.2.2 Náhodný počet stehů

Dalším z testovaných přístupů byl generátor náhodných čísel, které jsme použili k porovnání s reálným počtem stehů. Průměrně náhodný generátor správně detekoval stehy na osmnácti snímcích, čímž dosáhl přesnosti 13.67%. Tento výsledek naznačuje, že náhodný generátor není efektivním nástrojem pro detekci stehů v obrazech prasečích končetin a potvrzuje důležitost použití sofistikovanějších metod, jako jsou algoritmy založené na morfologických úpravách a neuronových sítích.

### 3.2.3 Konvoluční neuronové sítě

V této práci byla použita síť, která obsahuje tyto vrstvy:

2x	Conv2D	21 filtrů, (3,3) jádro, padding = 1, ReLu
1x	MaxPooling	(2,5) jádro, posun = 4, padding = 1
1x	BatchNormalization	—
1x	Conv2D	21 filtrů, (1,1) jádro, padding = 1, ReLu
1x	MaxPooling	(3,3) jádro, posun = 2, padding = 1
1x	BatchNormalization	—
1x	Conv2D	7 filtrů, (1,1) jádro, ReLu
1x	Flatten	—
1x	Dropout	(0.1)
1x	Dense	49, ReLU
1x	Dropout	(0.1)
2x	Dense	49, ReLu
1x	Dense	7, Softmax
	Tvar vstupu: 50,160,batch	Počet parametrů: 93 583

Tabulka 1: Architektura vlastního modelu



Při návrhu sítě jsme se snažili o co nejmenší a nejefektivnější síť, s ideálně malým počtem parametrů. Síť měla tendence učit se celý dataset nazpaměť, hlavně kvůli jeho omezené velikosti. Proti tomuto jevu zvanému overfitting jsme využili Dropout vrstvy, L2 regularizaci a augmentaci ve smyslu Flipování. Po těchto úpravách byla síť schopna dosáhnout až 71% na validačních datech. Příjemná vlastnost je, že pokud se síť dopustí chyby, byla vždy v rámci 1, maximálně 2 stehů.

## 4 Spuštění

### 4.1 Spustitelný skript

Pro spuštění souboru `run.py`, který se nachází ve složce `src` v repozitáři na GitHubu postupujte následovně:

1. Otevřete příkazový řádek nebo terminál.
2. Přejděte do složky, kde se nachází váš repozitář. Pokud je vaše složka `src` uvnitř repozitáře, použijte následující příkaz:

```
cd cesta/k/složce/src
```

3. Spusťte soubor `run.py` pomocí příkazu `python run.py`. Prvním argumentem je vždy cesta k výstupnímu souboru ve formátu CSV. Další argumenty mohou obsahovat přepínač `-v`, který spouští vizuální režim s debug obrázky, a názvy obrázků, které se snažíte detekovat.

```
python run.py výstupní_soubor.csv [-v] obrázek1.jpg obrázek2.png ...
```

4. Pokud chcete spustit vizualizaci, použijte příkaz s přepínačem `-v`:

```
python run.py výstupní_soubor.csv -v obrázek1.jpg obrázek2.png ...
```

Tímto způsobem spustíte skript `run.py` a provedete detekci obrázků podle zadaných parametrů.

### 4.2 Interaktivní notebook

Dále je k dispozici interaktivní soubor `main.ipynb`, který se nachází v repozitáři. Tento soubor umožňuje uživateli vyzkoušet další způsoby detekce. Všechny buňky jsou podrobně okomentovány, aby uživatel věděl, co dělají.

## 5 Závěr

V rámci této semestrální práce jsme provedli srovnání a evaluaci různých metod detekce stehů na prasečích končetinách, které byly použity při tréninku studentů. Na základě našich experimentů jsme zjistili, že nejlepších výsledků dosáhla konvoluční neuronová síť. Tato metoda se ukázala jako efektivní při identifikaci a lokalizaci stehů v obrazech a dosáhla vysoké přesnosti v porovnání s ostatními metodami.

Zároveň jsme zjistili, že morfologické úpravy, jako je dilatace, eroze, otevření a uzavření, jsou užitečnými nástroji pro zpracování obrazu a zlepšení výsledků detekce stehů. Při správném kombinování těchto operací jsme dosáhli značného zlepšení v porovnání s jednoduššími metodami, jako je například náhodný generátor čísel.

Celkově lze říci, že pro účinnou analýzu kvality stehů na prasečích končetinách je nejvhodnější přístup kombinace konvolučních neuronových sítí s morfologickými úpravami. Tento přístup poskytuje robustní a spolehlivé výsledky, které mohou být využity pro monitorování a hodnocení chirurgických výkonů.