

Bakalářská práce



České  
vysoké  
učení technické  
v Praze

**F3**

Fakulta elektrotechnická  
Katedra řídicí techniky

# Sledování pohybu zvířete na základě videozáznamu

Jan Machek

21. 5. 2014



České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra řídicí techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Jan Machek**

Studijní program: Kybernetika a robotika  
Obor: Systémy a řízení

Název tématu: **Sledování pohybu zvířete na základe videozáznamu**

Pokyny pro vypracování:

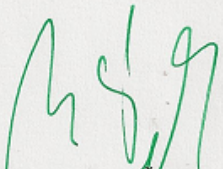
1. Seznamte se s metodami použitelnými pro sledování pohybu zvířete v omezeném prostoru na základe videozáznamu ze staticky umístěné videokamery.
2. Diskutujte a otestujte nejpoužitelnější metody z výše uvedeného bodu
3. Implementujte nástroje pro hromadné zpracování video záznamu
4. Výše uvedenými nástroji zpracujte dodané videozáznamy
5. Diskutujte výsledky a průběh zpracování, navrhnete případnou optimalizaci

Seznam odborné literatury:

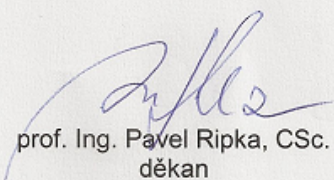
- [1] Sonka, M.; Hlavac, V.; Boyle, R.: Image Processing, Analysis, and Machine Vision. Toronto 2008.
- [2] Svoboda, T.; Kybic, J.; Hlavac, V.: Image Processing, Analysis, and Machine Vision: A MATLAB Companion. Toronto 2008.
- [3] Hlavac, V.; Schlesinger, M. I.: Ten lectures on statistical and structural pattern recognition. Dordrecht 2002.
- [4] Tillett, R. D.; Onyango, C. M.; Marchant, J. A.: Using model-based image processing to track animal movements. Computers and Electronics in Agriculture. 1997, vol. 17 (2), pp. 249-261.
- [5] Lind, N. M.; Vinther, M.; Hemmingsen, R. P.; Hansena, A. K.: Validation of a digital video tracking system for recording pig locomotor behaviour. Journal of Neuroscience Methods. 2005, vol. 143(2), pp. 123-132.

Vedoucí: Ing. Jakub Parák

Platnost zadání: do konce letního semestru 2014/2015

  
prof. Ing. Michael Šebek, DrSc.  
vedoucí katedry



  
prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 3. 1. 2014



## Poděkování / Prohlášení

Rád bych poděkoval MVDr. Monice Baxa, MVDr. Štefanovi Juhasovi a prof. MVDr. Janu Motlíkovi, DrSc. z Ústavu živočišné fyziologie a genetiky Akademie věd České republiky v Liběchově za poskytnutá data ke zpracování a děle MUDr. Jiřímu Klempířovi, PhD. z Neurologické kliniky a Centra klinických neurověd 1. lékařské fakulty University Karlovy v Praze a Všeobecné fakultní nemocnice v Praze za odbornou konzultaci ze strany Všeobecné fakultní nemocnice v Praze. Také bych rád poděkoval Ing. Matouši Pokornému z Katedry teorie obvodů Fakulty elektrotechnické Českého vysokého učení technického v Praze za konzultace a přínosné rady a připomínky. Za zprostředkování spolupráce s Katedrou teorie obvodů bych rád poděkoval jejímu vedoucímu prof. Ing. Pavlu Sovkovi, CSc. Největší poděkování patří vedoucímu mé bakalářské práce Ing. Jakubovi Parákovi za trpělivost, cenné rady a nápady a za vřelý přístup.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 21.5.2014.....

.....*Jm Machek*.....

## Abstrakt / Abstract

V presentované práci je popsána detekce a následné sledování pohybujícího se objektu – zvířete – na základě videozáznamu. Sledovanými zvířaty jsou transmutagenní prasata trpící Huntingtonovou chorobou. Je představeno, otestováno a vzájemně porovnáno několik metod sloužících k tomuto účelu. Nejlepší z navržených metod jsou následně použity k dávkovému zpracování videozáznamu. Pro pohodlnější a jednodušší zpracování videí je vytvořeno grafické uživatelské rozhraní v prostředí MATLAB. Získané výsledky ze všech dílčích částí práce jsou v závěru diskutovány.

**Klíčová slova:** zpracování obrazu, dávkové zpracování, segmentace, sledování pohybu, zvířecí modely

Algorithms for object detection and movement tracking are presented in this thesis. Tracked object is an animal model of disease. Animal models used for the research are transmutagen pigs suffering from Huntington's disease. Several methods are introduced, implemented and compared. The best fitting ones are then tested in detail and used for batch processing of recorded videos. A graphical user interface was created in MATLAB environment for easy and comfortable analysis of available videos. Acquired results are discussed in the end of this thesis.

**Keywords:** image and video processing, batch processing, segmentation, motion tracking, animal models

**Title translation:** Tracking animal movements from the video recording

## / Obsah

<b>1 Úvod</b> .....	1
<b>2 Motivace</b> .....	2
<b>3 Předšlé práce</b> .....	3
3.1 Zvířecí modely nemocí .....	3
3.2 Sledování prasete .....	3
<b>4 Popis a příprava videa</b> .....	5
4.1 Popis zpracovávaných videí .....	5
4.2 Dávková konverze .....	6
<b>5 Metody založené na separaci popředí a pozadí</b> .....	7
5.1 Metody se staticky definova- ným pozadím .....	7
5.1.1 Otsuova metoda .....	7
5.1.2 K-Means shlukování .....	10
5.1.3 Rozdělování a slučová- ní regionů .....	11
Růst regionů .....	11
Slučování regionů .....	12
Rozdělování a slučování .	12
5.1.4 Odečítání pozadí .....	13
5.2 Metody využívající průběžné modelování pozadí .....	14
5.2.1 Mediánový model po- zadí .....	14
5.2.2 Pozadí jako směs gaus- siánů .....	16
5.2.3 Implementace a disku- ze výsledků .....	17
<b>6 Sledování na základě optické- ho toku</b> .....	22
<b>7 Kernel-based sledování</b> .....	26
<b>8 Metody založené na geomet- rickém modelu popředí</b> .....	29
8.1 Proložení elipsou .....	29
8.2 Bodový model .....	30
<b>9 Porovnání metod</b> .....	36
<b>10 Hromadné zpracování videa</b> .....	41
10.1 Implementace .....	41
10.2 Klasifikace pohybu .....	41
<b>11 Diskuze výsledků</b> .....	44
<b>12 Závěr</b> .....	45
Literatura .....	46
<b>A Obsah přiloženého CD</b> .....	49
<b>B Ukázka klasifikovaného pohy- bu pro jednotlivé dny</b> .....	50





# Kapitola 1

## Úvod

Videozáznam je jedním z nástrojů pro sledování pohybu zvířat. Velkou výhodou je neinvazivnost takovéto metody. Sledované zvíře nemusí nosit žádný fyzický senzor.

Porovnáním záznamů zaznamenaných v určitém časovém rozmezí lze odhadnout změny v chování sledovaných objektů. Toho se využívá při zkoumání průběhů nemocí či vedlejších účinků medikace. Dlouhodobým sledováním zvířat s pořizováním videozáznamů lze analýzou těchto nahrávek určit změny v denních rytmech, jako je například délka spánku, intenzita příjmu potravy nebo pohybová aktivita.

K účelu sledování objektů ve videozáznamu byla v minulosti představena řada metod. Sledování pohybu zvířete může být založené na předem známém modelu, který se algoritmus snaží co nejpřesněji umístit, na hledání takového objektu v obraze, který vyhovuje určeným kritériím, nebo určení pohybujícího se objektu ve scéně pomocí modelování pozadí. Úkolem této práce je nevhodnější metody implementovat a vzájemně porovnat.

Na videozáznamech uvažovaných v této práci jsou nahrána prasata při pohybu ve svých kotcích. K nahrávání byla použita statická kamera sledující kotec shora. Mezi sledovanými zvířaty jsou zdravé kusy i kusy trpící Huntingtonovou nemocí. Nemocné kusy se někdy označují jako tzv. zvířecí modely nemoci.

Huntingtonova nemoc je dědičné neurodegenerativní onemocnění mozku, které se u lidí projevuje nekoordinovanými trhavými pohyby končetin i celého těla. V průběhu nemoci se také snižuje mentální zdraví a kognitivní schopnosti nemocného člověka.

Analýza dlouhodobého záznamu pohybů prasat s Huntingtonovou chorobou přispěje k poznání vlivu této nemoci na prasata a k pochopení průběhu této choroby.

K analýze zaznamenaných dat je kvůli délce záznamů a jejich komplexnosti potřeba strojového zpracování, které je tématem následující práce.

## Kapitola 2

### Motivace

Cílem představené práce je výběr vhodných metod pro sledování objektu a jejich následná implementace. Všechny uvažované metody mají za cíl urychlit a zjednodušit analýzu videozáznamů pohybu zvířete ze statické kamery. Porovnávané metody budou blíže popsány, implementovány a použity přímo na dostupných datech.

Vzhledem k počtu a k celkové délce nahraných videí bude také zapotřebí navrhnout metodu pro dávkové zpracování a následnou automatickou analýzu velkého množství dat bez nutnosti zásahu člověka. Dávkové zpracování bude zahrnovat konverzi dostupných videozáznamů do formátu, ve kterém je bude později možné zpracovat v prostředí MATLAB. Konvertovaná videa budou následně opět po dávkách analyzována nejvhodnějšími metodami, které vyplynou z porovnání představených metod. Časové a hardwarově náročná konverze a analýza videí by tak mohla probíhat v takových částech dne, kdy není výpočetní výkon tolik využíván, např. v noci.

Pro zpříjemnění práce s navrženými metodami bude připraveno grafické uživatelské rozhraní v prostředí MATLAB, které bude umožňovat nastavení různých metod a různých parametrů konverze či vlastní analýzy poskytnutých dat. Toto rozhraní bude také umožňovat snazší výběr videí ke zpracování.

Výsledek takto zpracovaných videí bude vypovídat o pohybu zvířete v uzavřeném prostoru. Části dne, které jsou pro analýzu nejzajímavější, jsou ráno, období oběda a období večere. Proto bude do grafického uživatelského rozhraní přidána možnost výběru specifické denní doby, podle které budou vybrána odpovídající videa.

Další důležitý aspekt je celkový pohyb za určité období. Jedním z předpokládaných vlivů Huntingtonovy choroby na nemocné zvíře by měla být změna v množství deního pohybu. Mělo by tedy ze zpracovávaných videí jít určit, jestli se průměrná aktivita prasete přes celý den zvětšuje, zmenšuje či zůstává na přibližně stejné hladině.

Data získaná analýzou nahraných videozáznamů poslouží k rozpoznání vlivu průběhu Huntingtonovy nemoci na chování sledovaných zvířat. Díky dlouhodobému sledování zvířat bude možné porovnat změny jejich denních rytmtů a celkové aktivity a pohyblivosti.

# Kapitola 3

## Předešlé práce

### 3.1 Zvířecí modely nemocí

Zvířecí modely jsou v medicíně hojně používány ke studiu nemocí, strategiích léčby a testech medikace ve fázích před klinickými studiemi. Prasačí zvířecí model byl využit např. Andersonem a kol. [1] při zkoumání vlivu diazepamu na psychiku prasat. Studie byla zaměřena na změny vnímání strachu a úzkosti. Prasata pod vlivem diazepamu podstoupily testy obavy z otevřených prostor nebo reakci na světlo a tmu. Mikkelsen a kol. [2] použili model prasete pro vyvolání Parkinsonovy choroby a sledování jeho průběhu.

Rubinsztein publikoval studii [3], ve které využil zvířecí modely ke studiu Huntingtonovy choroby. V jeho experimentech byly použity myši, které mají výhodu krátkého životního cyklu, uniformní genetiky a stejné mutace Huntingtonovy nemoci, kterou můžeme najít i u lidí. Pokud je požadavkem extrémně krátký životní cyklus, zmiňuje Rubinsztein použití octomilek. Přestože jsou octomilky hmyz a ne savci jako myši či prasata, Huntingtonova choroba se u nich projevuje podobně jako u lidí. Prasečí zvířecí modely byly při studiu Huntingtonovy choroby použity Baxa a kol. [4]. Zde je výhodou naopak dlouhý životní cyklus a vysoká hmotnost zvířete v porovnání s výše zmíněnými zvířecími modely.

### 3.2 Sledování prasete

Videozáznam jako nástroj pro sledování pohybu zvířete používá např. Kongstro, který se zaměřil na analýzu chůze prasete [5]. Pochodující zvířata jsou nahrávána kamerou umístěnou nad nimi a z jejich siluety je poté klasifikována nemohoucnost nebo slabost nohou. Pro segmentaci prasat bylo využito prahování Otsuovou metodou.

Tuto metodu segmentace využívá také Kashiha [6]. Ten se ve své práci zabývá monitorováním aktivity prasat v kotcích. Prasata nejsou na rozdíl od našeho případu v kotci po jednom, ale sdílejí jeden kotec ve více kusech. Pro analýzu pohybu se využívá proložení segmentovaných dat několika elipsami. Každému praseti patří jedna elipsa.

Lind a kol. [7] používají k segmentaci metodu odečítání obrazů. Na jednom snímku je pouze pozadí a tento snímek se odečítá od aktuálního snímku kamery, na kterém je i sledovaný objekt. Tím je získán rozdíl mezi těmito obrázky – tedy sledovaný objekt. Tato metoda je výhodná oproti prahování v případě, že sledovaný objekt není barevně či intenzitou tak výrazně odlišný od pozadí. Znatelná odlišnost zvířete a pozadí však stále musí zůstat.

Pokročilá metoda analýzy je předvedena Marchantem a Onyango [8]. K přesnému sledování prasete je použit tzv. point-distribution model, model tvaru prasete složený z bodů spojených křivkami. Výhodou tohoto přístupu je fakt, že lze sledovat nejen translaci a rotaci prasete jako je to u prokládání prasete elipsou zmíněného výše, ale také lze pozorovat prohýbání prasete. K získání konkrétního tvaru modelu bodů v jednotlivých snímcích je využito posuzovací funkce navržené pro tento konkrétní případ.

Stejnou metodu sledování pohybu, tentokrát na příkladu sledování pohybu lidí, detailně popisuje Baumberg [9].

Matematickým popisem takových modelů a jejich trénováním se zabývá Cootes [10] na příkladu modelu resistoru. V článku je popsáno získání průměrného tvaru resistoru a je předvedeno, jak získat jednotlivé parametry modelu jako je translace či rotace.

Comanciu [11] a Zhang [12] se zabývají tzv. kernel-based sledováním objektu v obrazu. Tato metoda vychází z předpokladu, že poloha (případně i orientace) sledovaného objektu je v daném snímku známa. Po prozkoumání okolí této známé polohy je poté nalezena poloha (a orientace) nová a z té je vycházeno v dalším snímku.

Další metody segmentace obrazu a sledování objektu jsou představeny v knize Image Processing, Analysis and Machine Vision [13].

Z výše jmenovaných zdrojů bylo v této bakalářské práci vycházeno.

# Kapitola 4

## Popis a příprava videa

### 4.1 Popis zpracovávaných videí

Pohyb prasat v kotci byl zaznamenán statickou kamerou umístěnou nad kotcem. Prasata se pohybují pouze v prostoru jejich kotce. Záznamy byly pořízeny softwarem Vivotek st2403 a výstupní formát je .avi. Použitý software ale kóduje video vlastním kodekem, který není kompatibilní s MATLABovskými distribucemi, a všechna videa proto musela být konvertována do standardního formátu .avi.

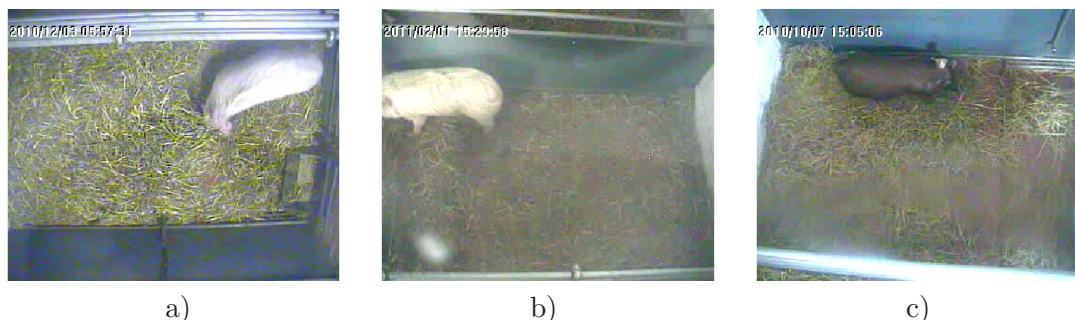
Rozlišení všech nahraných videí je 352x288 pixelů. Vzorkovací frekvence je pro všechna videa 24 snímků za vteřinu. Nahraná videa pro všechna prasata mají vždy okolo 5 minut ( $\pm 10$  vteřin) a navazují na sebe. Jejich celkový počet je 11 229.

Celkem byla nahrávána 3 prasata – Alf, Babe a Bruno. První dvě jmenovaná jsou bílá, třetí je hnědá. Prase Alf bylo nahráváno každý den od 5:30 ráno ( $\pm 10$  minut) do 6:00 rána a poté znovu od 11:00 dopoledne od 16:25 odpoledne. Prasata Babe a Bruno byla nahrávána každý den od 5:30 ráno ( $\pm 10$  minut) do 7:00 ráno a poté znovu od 12:00 dopoledne od 16:30 odpoledne. Ne každý den však byla nahrána všechna videa a tak není sekvence všech videí kompletní. V některých dnech je přeskočeno dopolední nahrávání, jindy zase odpolední nahrávání začíná či končí později resp. dříve.

Každé video má ve svém názvu zakódováno číslo kotce, jméno prasete a den a čas začátku videa. Příkladem názvu videa je *alf\_2011020115300101*. Toto video je záznam prasete Alfa z 1. 2. 2011 a začíná v 15:30:01, který byl nahrán v kotci č.1.

Kotec byl ohraničen ze všech čtyř stran a to buď pevnou stěnou nebo plechovým předělem. Prasata jsou krmena přímo v kotcích.

Ukázky jednotlivých snímků z jsou zobrazeny na následujícím obrázku č. 4.1.



**Obrázek 4.1.** Ukázky snímků z videozáznamů. Na obr. jsou ukázky ze záznamů a) prasete Babe ráno, b) prasete Alf odpoledne a c) prasete Bruna odpoledne.

## 4.2 Dávková konverze

Konverze videa do standardního formátu .avi byla provedena ve freeware programu VirtualDub verze 1.10.4<sup>1)</sup>. Tento software je specializován na práci právě s videi ve formátu .avi.

Pro samotnou konverzi byly napsány dva .bat skripty. První skript pouze pro každé video vygeneroval .vcf soubor s adresou vstupního videa, adresou výstupního videa a parametry konverze. Soubory .vcf jsou instrukční soubory pro VirtualDub, podle kterých je provedena konverze. Druhý .bat skript sloužil pro rekurzivní prohledání složky se vstupními videi, pro vygenerování výše popsaných .vcf souborů spuštěním skriptu prvního a spuštění VirtualDubu s instrukcemi podle těchto .vcf souborů.

Bylo testováno také převedení videa na sekvenci jednotlivých .jpg nebo .png obrázků. Ukázalo se ale, že taková konverze je pomalá a navíc paměťově velmi náročná, protože pro každé video musí být vygenerováno průměrně 7200 obrázků odpovídajícím jednotlivým snímkům.

Konverze všech videí do standardního .avi probíhala na vzdálené ploše na počítači s velkokapacitním diskem a trvala přibližně 40 hodin. Velikost každého videa se konverzí zvětšila téměř 2.5x, kvalita ale zůstala stejná.

---

<sup>1)</sup> <http://www.virtualdub.org/>

# Kapitola 5

## Metody založené na separaci popředí a pozadí

Nejintuitivnější přístup ke sledování pohybu objektu ve scéně je rozdělit určitou metodou obraz na pozadí a popředí. Segmentace obrazu je metoda, při níž je obraz pokud možno automaticky rozdělen na segmenty (neboli části či regiony) podobných vlastností. V případě zde prezentovaných videí se pod segmentací obrazu rozumí oddělení popředí od pozadí, tedy oddělení prasete od podlahy a zbytku kotce. V této kapitole jsou popsány používané segmentační metody, které byly také implementovány a na dostupných videích prasat otestovány.

Použití této metody pro sledování pohybu prasete je prosté. Každý snímek videa je segmentován a pozice prasete je určena jako střed regionu, který zvířeti v obraze odpovídá.

### 5.1 Metody se staticky definovaným pozadím

V této sekci budou představeny metody, které nemění model, se kterým při segmentaci pracují. Je tedy jedno, od jakého snímku je metoda spuštěna a nebo kolik snímků je zapotřebí segmentovat.

#### 5.1.1 Otsuova metoda

Otsuova metoda segmentace [14] byla uvedena v roce 1975 jako nová možnost segmentace černobílého obrazu. Výhodou této metody je nezávislost na apriorní znalosti obrazu a také velká rychlost.

Vstupním parametrem je obraz v šedém spektru a výstupem jsou dvě množiny pixelů. V ideálním případě jsou těmito množinami pozadí a popředí. Rozdělení obrazu do uvažovaných dvou množin probíhá metodou prahování a jako práh je použita hodnota vypočítána algoritmem, který navrhl právě Otsu ve své práci.

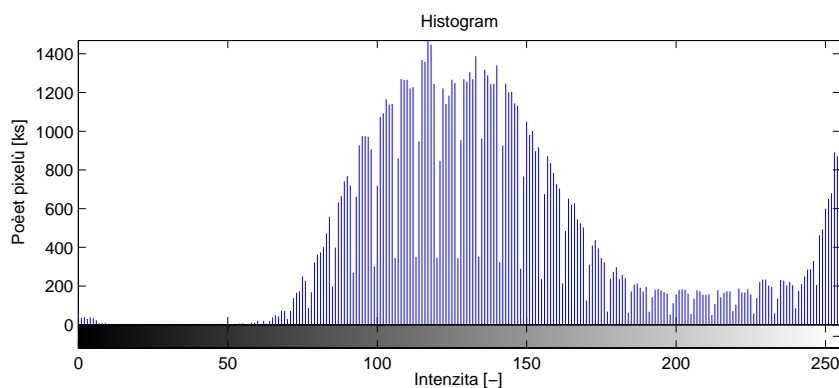
Základní strukturou používanou k určení prahu u metod segmentace prahováním je histogram. Histogram je grafické znázornění rozdělení intenzit jednotlivých bodů obrazu do předem daného počtu intervalů tzv. binů. Nejlepší situace pro určení prahu je taková, při které je v histogramu uvažovaného obrazu možné pozorovat dvě špičky oddělené hlubokým a čistým údolím. Prah je v takovém případě zvolen uprostřed předělu mezi dvěma výraznými špičkami. Problém nastává, když se pozadí a popředí promíchávají a histogram tak není jednoznačně oddělen.

V případě uvažovaných videí je střed bílého prasete za všech denních i umělých osvětlení, při kterých byla prasata nahrávána, možné v histogramu rozlišit poměrně dobře. Intenzita pixelů ve středu hřbetu prasete je vždy velmi vysoká proti zbytku a tvoří v histogramu úzkou špičku. Na bocích, hlavě a zadní části prasete už ale není intenzita tak vysoká a přechod mezi zvířetem a pozadím je v histogramu smíchan s dalšími body mimo prase, např. s přechodem mezi stěnou a podestýlkou.

Histogram se navíc mění a posouvá i se změnou intenzity osvětlení. Prah nelze tedy nastavit na pevně danou hodnotu, ale musí být adaptabilní.



**Obrázek 5.1.** Jeden vybraný snímek videozáznamu a snímek po barevné transformaci do šedého spektra.



**Obrázek 5.2.** Histogram snímku z obr. č 5.1 b).

Jeden snímek videozáznamu spolu s jeho transformací do šedého spektra je vidět na obr. č. 5.1 . Na obr. č. 5.2 je zobrazen histogram snímku z obr. č 5.1 b).

Barevná transformace byla provedena příkazem `rgb2gray` v prostředí MATLAB. Tento příkaz používá jednoduchý přepočítání mezi barevnými kanály a intenzitou v šedém spektru podle vztahu

$$I = 0.2989 * R + 0.5870 * G + 0.1140 * B, \quad (5.1)$$

kde  $I$  je výsledná intenzita v šedém spektru,  $R$  je intenzita v červeném kanálu,  $G$  intenzita v zeleném kanálu a  $B$  je intenzita v kanálu modrém.

Výběr prahu podle Otsuovy metody se zakládá na matematické statistice.

Zpracovávaný obraz je reprezentován pixely (šedého spektra) v  $L$  intervalech intenzity. Počet pixelů v intervalu  $i$  je  $n_i$  a  $N$  je celkový počet pixelů. Na rozdělení pixelů do intervalů se lze poté dívat jako na rozdělení pravděpodobnosti. Pravděpodobnost výskytu pixelu v určitém intervalu je

$$p_i = \frac{n_i}{N}. \quad (5.2)$$

Hledaný práh označíme jako  $k$ . Tímto prahem lze pixely rozdělit do dvou množin:  $C_0$ , do které spadají intervaly  $[1, \dots, k]$ , a  $C_1$  s intervaly  $[k + 1, \dots, L]$ . Pravděpodobnost výskytu jednotlivých množin je



$$\omega_0 = P(C_0) = \sum_{i=1}^k p_i = \omega(k), \quad (5.3)$$

$$\omega_1 = P(C_1) = \sum_{i=k+1}^L p_i = (1 - \omega(k)). \quad (5.4)$$

Střední hodnoty v jednotlivých množinách odvozuje Otsu jako

$$\mu_0 = \frac{\mu(k)}{\omega(k)}, \quad (5.5)$$

$$\mu_1 = \frac{\mu_T - \mu_k}{1 - \omega(k)}, \quad (5.6)$$

kde  $\mu(k)$  je kumulativní moment prvního řádu daný vztahem

$$\mu_k = \sum_{i=1}^k ip_i \quad (5.7)$$

a  $\mu_T$  je celková střední hodnota vyjádřená jako

$$\mu_T = \sum_{i=1}^L ip_i. \quad (5.8)$$

Otsu ukazuje, že hledaný práh  $k^*$  je takový práh, který maximalizuje hodnotu  $\sigma_B^2$  danou vztahem

$$\sigma_B^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2, \quad (5.9)$$

tedy

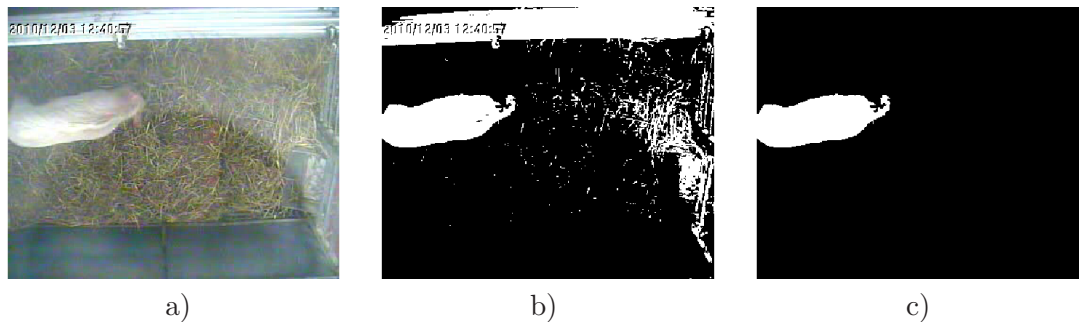
$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k). \quad (5.10)$$

Otsuova metoda pro nalezení vhodného prahu je implementována jako funkce v prostředí MATLAB a to v balíku Image Processing Toolbox pod názvem *graythresh*.

Prase je v segmentovaném obraze lokalizováno pomocí navržených heuristik. Nejdříve jsou odstraněny malé regiony, které se vyskytují v důsledku zašumění obrazu. Protože kamera je statická a kotce se po celou dobu nahrávání nemění, je možné obraz zmenšit o plochu horní stěny promítnuté do obrazu. Tato stěna má na obrázku vysokou intenzitu, je tak segmentována do stejné třídy jako prase. Prase se přes horní (z pohledu kamery) stěnu nemůže dostat, nepřeleze ji, není třeba ji uvažovat ve stejné třídě jako prase.

Výsledek prahování za pomoci Otsuovy metody pro nalezení vhodného prahu a následného použití výše uvedené heuristiky pro odstranění nevhodných objektů ze stejné třídy, ve které je i prase, je vidět na následujících obrázcích. Vypočítaný práh byl v tomto případě  $k^* = 0.651$  (hodnota je normalizována v intervalu intenzit  $< 0, 1 >$ , který odpovídá intervalu  $< 0, 255 >$ , tedy po přepočtu  $k^* = 166$ ).

Obě bílá prasata jsou pomocí výše popsaného postupu rozpoznána správně. Při testech na více vídeích se ukázala i nezávislost na světelných podmínkách. Ta je dána velkým rozdílem mezi světelnou intenzitou pozadí a světelnou intenzitou zvířete.



**Obrázek 5.3.** Výsledky detekce prasete. a) Zpracovávaný snímek, b) snímek po prahování s prahem vypočítaným podle Otsuovy metody, c) výsledek po použití heuristiky.

### 5.1.2 K-Means shlukování

Segmentace popsaná v předešlé sekci této kapitoly dokázala spolu s použitou heuristikou správně detekovat pozici zvířete ve snímku. Taková segmentace funguje ale pouze na objekty, které jsou výrazně jasnější nebo naopak tmavší než je pozadí. Předešlou metodu nelze použít k detekci hnědého prasete.

Další používanou metodou pro segmentaci obrazu je shlukování k-Means. K-Means shlukování je heuristický algoritmus, který se snaží rozdělit vstupní data do  $k$  shluků, ve kterých je vzdálenost každého bodu shluku od středu tohoto shluku minimální. Počet shluků musí být před začátkem algoritmu znám.

K-Means shlukován funguje podle následujícího algoritmu:

1. Náhodně se inicializuje  $k$  středů.
2. Body v prostoru se přidělí do shluků. Vždy je zvolen ten shluk, k jehož středu má uvažovaný bod nejbližší.
3. Jsou přepočítány polohy středů. Nová poloha je dána aritmetickým průměrem poloh bodů shluku.
4. Pokračuje se bodem 2, dokud není dosaženo lokálního minima – již se nezmenšuje vzdálenost bodů shluků od středů.

Problémem k-Means shlukování může být uváznutí v lokálním minimu při nevhodném počátečním umístění středů. To se dá vyřešit opakovaným spouštěním algoritmu.

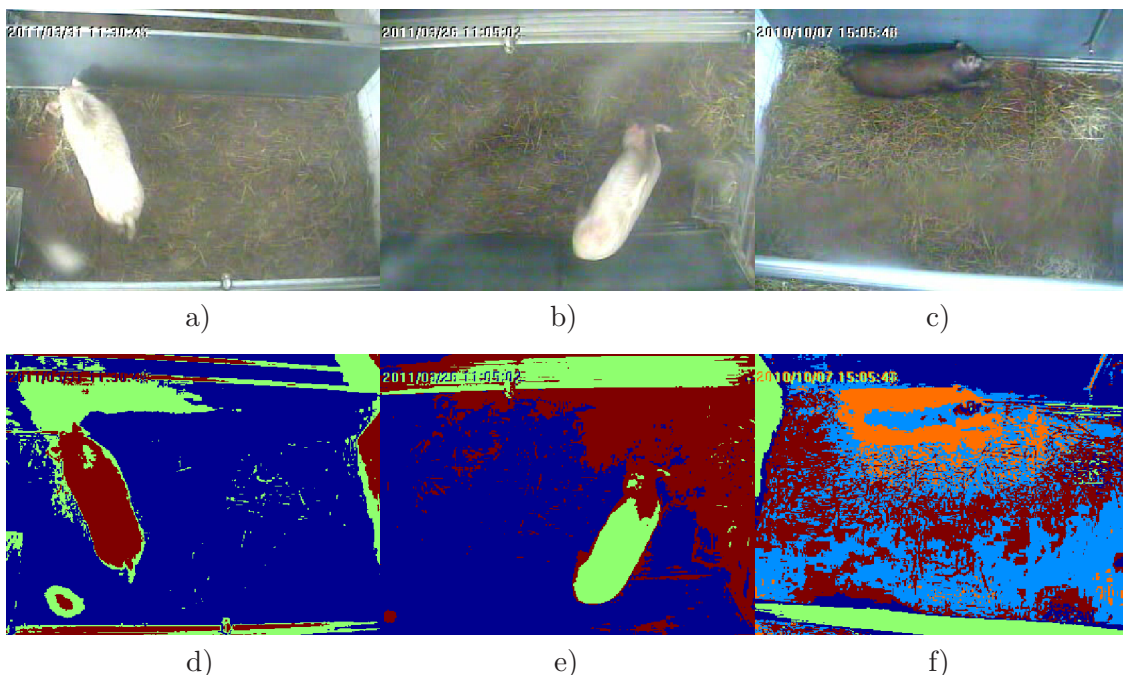
Při segmentaci obrazu jsou vstupem k-Means algoritmu jednotlivé barevné kanály. Intenzita v každém kanálu poté udává souřadnice jednotlivých bodů a v tomto prostoru jsou inicializovány počáteční středy. Vzdálenost jednotlivých bodů od středů je v průběhu iterací počítána jako Euklidovská.

Shlukování k-Means je již obsažené v MATLABu v balíku Statistics toolbox.

Na obrázku č. 5.4 jsou prezentovány výsledky segmentace k-Means shlukování. V horním řádku jsou originální snímky a ve spodním řádku jsou zobrazeny výsledné segmentace. U prasat Alf a Babe (obrázky a) a b) ) byly uvažovány 3 shluky u hnědého prasete Bruna 5 shluků. Pro menší počet shluků splývá barva prasete s barvou podestýlky v kotci a shluk této barvy je pak téměř přes celou plochu podestýlky s prasetem a není možné v něm zvíře identifikovat.

Segmentace s využitím shlukování k-Means již na rozdíl od prahování obrazu v šedém spektru dokáže rozpoznat hnědé prase. Region náležící hnědému praseti ale pokrývá pouze malou část jeho plochy a také není zvířeti tvarově podobný. Problémem je také potřeba určit počet shluků při změně podmínek, např. světelných.

Už zde se ukazuje velký problém nahraných videí hnědého prasete Bruna a to přílišná rozostřenost v dolní části videa. Kamera v těchto místech nejspíše zachytávala jak kotec,



**Obrázek 5.4.** Segmentace k-Means shlukováním. Obrázky a), b) a c) jsou originální snímky z videozáznamu, obrázky d), e) a f) jsou výsledky segmentace.

tak i odlesk umělého osvětlení. Kvalita videa je přibližně od poloviny výšky až do spodku videa a přes celou vodorovnou osu velmi špatná.

Bílá prasata jsou opět rozpoznána správně.

### 5.1.3 Rozdělování a slučování regionů

Metoda rozdělování a slučování je metodou přímé konstrukce regionů podobných vlastností.

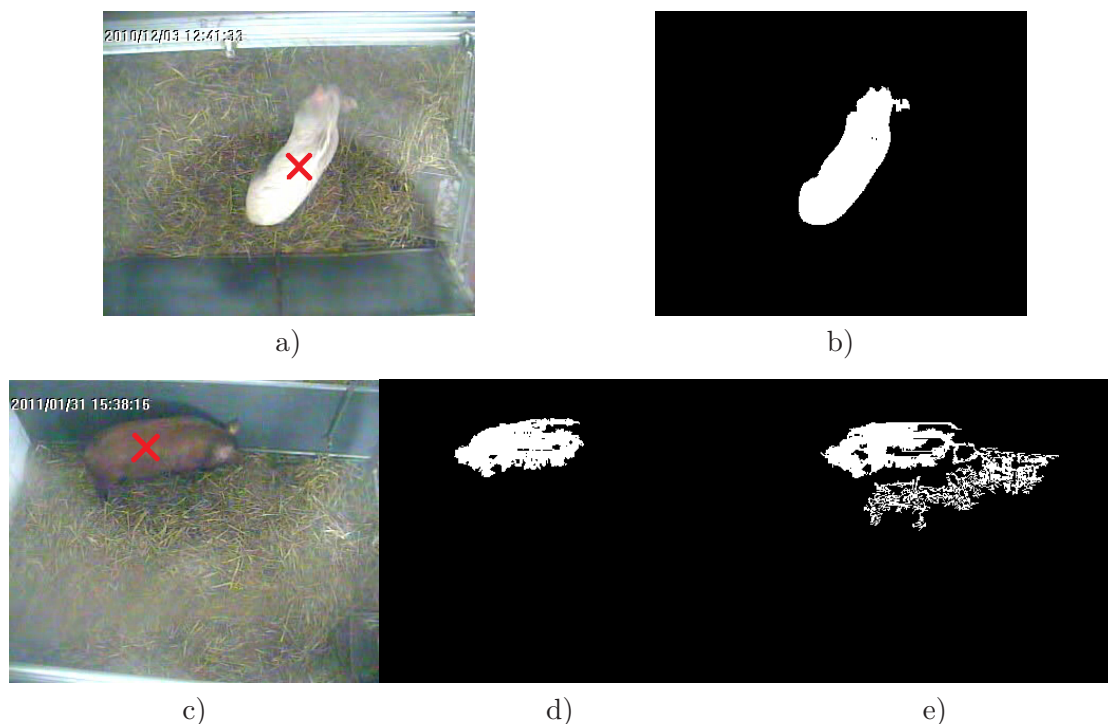
#### Růst regionů

Nejjednodušším přístupem konstrukce regionů je jejich růst (anglicky region growing). Algoritmus je velmi jednoduchý. Každý region je konstruován z malého předem zadaného regionu tak, že se daný region slučuje s pixely či celými regiony se zadaným regionem sousedícími a splňujícími zadané vlastnosti. Takto se přidávají všechny sousední pixely resp. regiony až dokud již nejsou žádné sousedící regiony, které by splňovaly zadanou podmínku.

Při růstu si region musí stále zachovávat homogenitu ať už ve smyslu intenzity barvy, tvaru či jinou.

Jako kritérium homogenity byla v této práci použita intenzita jednotlivých pixelů regionu. Předem stanovený region (stačí i jeden pixel a náhodné další v jeho okolí) se zvětšoval o takové své sousední pixely, které měly intenzitu v předem zadaném okolí. Výsledky lze vidět na obrázku č. 5.5.

Z obrázku č. 5.5 je vidět, že metoda je velmi úspěšná pro bílé prase a toto prase je velmi přesně segmentováno. U hnědého prasete už není segmentace příliš přesná. Na obrázku č. 5.5 e) je navíc předvedeno nebezpečí použité této metody při segmentaci hnědého prasete. Při špatně zvoleném počátečním regionu (na obr. č. 5.5 e) je poč. region jiný než na obr. č. 5.5 c), ale pouze ve vzdálenosti jednotek pixelů) se může stát, že se do regionu, který by měl odpovídat pouze praseti přimíchají i pixely podestýlky, které mají podobnou barvu jako prase a poté se tato chyba rozšíří.



**Obrázek 5.5.** Regiony vygenerované pomocí metody růst regionů. První řádek výsledky pro prase Alf, ve druhé řádku jsou výsledky pro prase Bruno. a), c) Zadné počáteční regiony, b), d) výsledné regiony, e) region pro ukázkou chyby metody.

S možností řešení tohoto problému přichází následující metoda.

### ■ Slučování regionů

Základní myšlenka metody slučování regionů je rozdělení obrazu do určitého počtu malých regionů a ty poté slučovat podle kritérií homogenity. Protože počáteční rozdělení obrazu nerozdělí celou plochu do jednotlivých pixelů, dá se tedy předpokládat, že se tímto vyřeší problém prolínání prasete a pozadí pouze přes několik pixelů, jak tomu bylo u minulé metody v případě hnědého prasete.

Pro segmentované regiony  $R$  musí platit následující dvě rovnice ([13] str. 224)

$$H(R_i) = PRAVDA, \quad i = 1, 2, \dots, S, \quad (5.11)$$

$$H(R_i \cup R_j) = NEPRAVDA, \quad i \neq j, \quad R_i \text{ je sousedem } R_j, \quad (5.12)$$

kde  $S$  je celkový počet regionů a  $H$  je funkce určující, zda je region homogenní. Sloučené regiony tedy musí všechny být homogenní a maximální, což znamená, že by už homogenními nezůstaly, pokud by se sloučily s některým ze svých sousedů.

Rozdělení celého obrazu do malých regionů a jejich následné slučování ale není praktické, protože prase zabírá pouze určitou část obrazu a zbytek je po částech relativně homogenní pozadí. Výhodnější metodu rozdělení obrazu přináší vylepšení popsané v následující sekci.

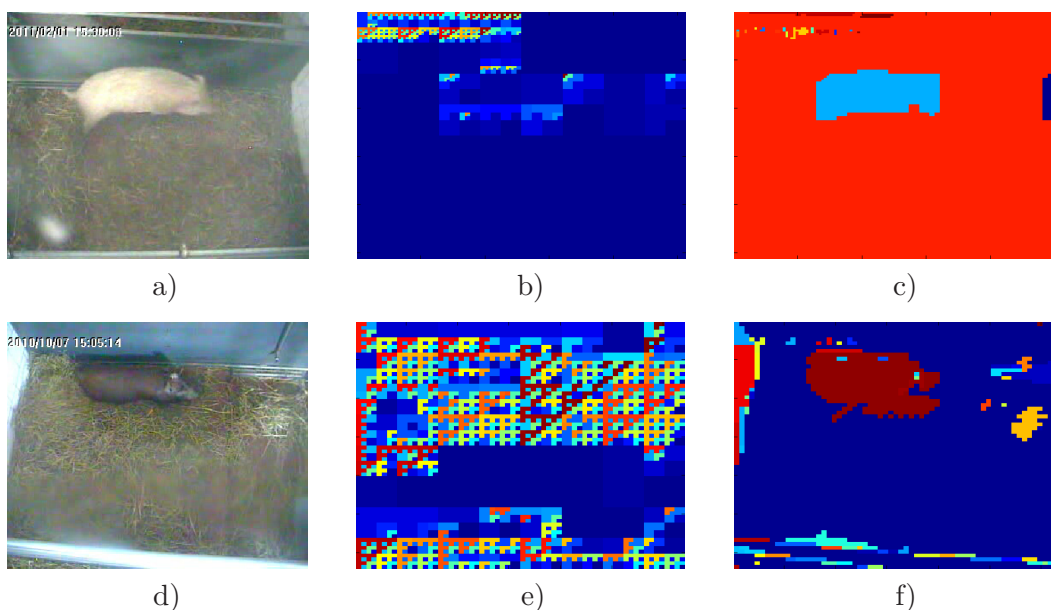
### ■ Rozdělování a slučování

Pro zachování co největších prvotních regionů v homogenních oblastech se využívá řízeného rozdělování obrazu.

Každý region, který není homogenní a nespĺňuje tedy rovnici (5.11), je rozdělen na čtyři stejně velké regiony. K určení homogenity posloužil v předvedeném případě rozptyl intenzity. Takto jsou zpracovány všechny regiony (včetně nově vytvořených), dokud všechny regiony rovnici (5.11) už nespĺňují. Rozdělování začíná s jediným regionem – celým obrazem.

Po rozdělení celého obrazu může začít slučování sousedících regionů ke splnění rovnic (5.11) a (5.12). Sloučení proběhne, pokud je rozdíl intenzit sousedících regionů menší než předem zadaný práh.

Výsledky předvedené metody jsou vidět na obrázku č.5.6. Výsledné obrázky jsou čtvercové. Zpracovávaný snímek byl transformován na čtverec kvůli jednoduššímu dělení.



**Obrázek 5.6.** Regiony vygenerované pomocí metody rozdělení a sloučení regionů. První řádek zobrazuje výsledky pro prase Alfa, ve druhé řádce jsou výsledky pro prase Bruna. a), d) aktuální snímky, b) e) rozdělení do homogenních regionů a c), f) sloučené regiony

Z výsledků je vidět, že jak bílé tak hnědé prase jsou v obrazu poměrně dobře rozpoznány. Problémem by bylo, pokud by se ve snímcích vyskytoval jiný homogenní region (např. větší část stěny), který by byl přibližně stejně velký jako prase. Poté by mohlo dojít k záměně. Pro odstranění regionů, které nejsou prase, lze použít podobnou heuristiku jako v minulých metodách – zaměřit se pouze na regiony určitých velikostí.

#### 5.1.4 Odečítání pozadí

V další často používané metodě segmentace a sledování pohybu je využito toho, že kamera snímající kotec shora je statická a zabírá stále stejné zorné pole. Lze tedy předpokládat, že kdyby byl získán obraz pozadí bez prasete, po odečtení tohoto obrazu a aktuálního snímku z kamery by zůstal pouze pohybující se objekt – prase.

Přesně to bylo se zdrojovými videi provedeno. Pro každé prase byl získán obrázek pozadí sloučením dvou částí obrazu z dvou různých snímků z videosekvence získané během určitého dne. Takto bylo získáno více snímků pozadí, vždy z jiného dne.

Dále bylo využito znalosti rozsvícení světel nad kotci. Vždy ráno od 5:00 do 10:00 a odpoledne přibližně od 16:00 byla intenzita osvětlení díky umělému svícení vyšší. Proto byly všechny dny uvažovány dva modely pozadí. Každý pro jinou intenzitu osvětlení.

Výsledné pozadí vzniklo průměrováním všech získaných pozadí dané intenzity ve stejnou denní dobu.

Experimentálně bylo zjištěno, že malé změny intenzity osvětlení v průběhu dne nejsou pro tento jednoduchý model pozadí velkým problémem.

Výsledky této segmentace jsou vidět na obr. č. 5.7. Po odečtení se obraz pomocí velmi nízkého prahu převádí na obraz binární. Bílá prasata jsou v rozdílovém obrazu vidět velmi dobře, u hnědého prasete je na rozdílový obraz ještě použito zvětšení kontrastu pomocí MATLABovského příkazu *imadjust*. Malé regiony jsou z odečteného obrázku odstraněny, tak je získán pouze region prasete.

Předvedené snímky z obr. č. 5.7 pocházejí z jiného dne, než ve kterých byly pořízeny snímky k vytvoření modelu pozadí. Z výsledků je vidět, že pro bílá prasata tato metoda funguje velmi dobře. Největší výhodou je jednoduché odstranění ostatních objektů stejné barvy z obrázku. Toho si lze všimnout u prasete Alf v levém sloupci. Zeď v pravé části snímku se intenzitou podobá samotnému praseti. Tím, že je ale obsažena i v modelu pozadí a v čase se její intenzita téměř nemění, se však odečte a z výsledného obrazu úplně zmizí.

Výhodou této metody je také fakt, že po segmentaci zůstane velmi dobře zachován tvar prasete.

Tato metoda je ale velmi špatně použitelná při zpracovávání videí s hnědým prasetem. Prase je přibližně rozpoznáno a detekováno. Problém je ale znatelný šum a velký odlesk ve spodní části videa, který se v průběhu celého nahrávání velmi měnil. Nebylo tak možné sestavit model pozadí, který by tento problém překonal.

Při zpracovávání všech videí touto metodou byla zjištěna velmi vysoká přesnost u bílého prasete Alf v průběhu celého dne. U druhého prasete Babe se ale ukázalo, že bylo v průběhu nahrávání videozáznamů několikrát manipulováno s kamerou. K pokrytí těchto manipulací je třeba zaznamenat všechny okamžiky této manipulace a získat model pozadí pro každou natavenou polohu kamery v průběhu sledování prasete. To je ale velmi nepraktické a pro automatizovaný provoz s minimalizací zásahů člověka nevhodné. I pro segmentaci prasete Babe od pozadí však metoda odčítání pozadí fungovala velmi dobře.

Implementované řešení využívající aktualizování modelu pozadí je předvedeno v následující kapitole.

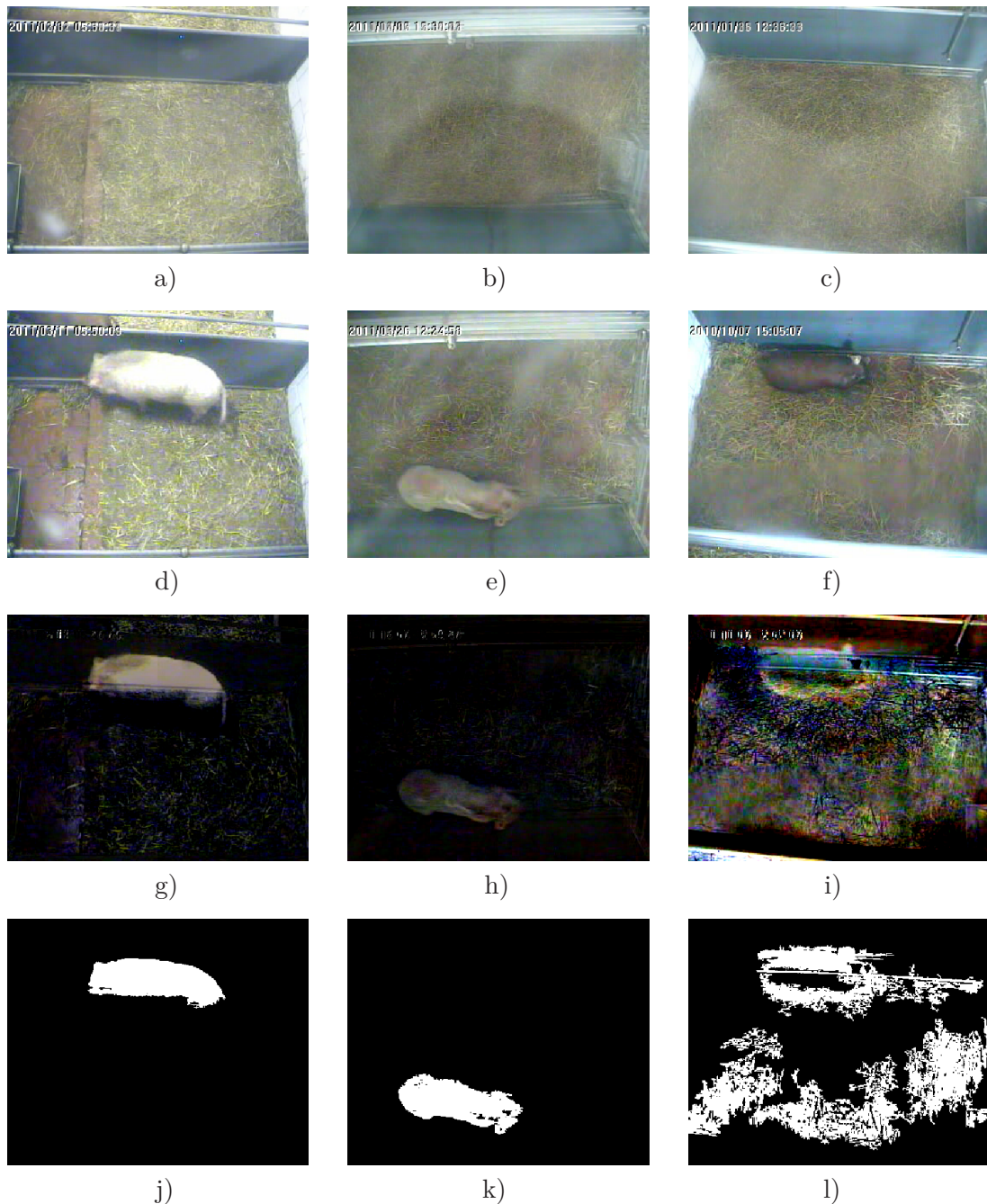
## 5.2 Metody využívající průběžné modelování pozadí

V sekci Odčítání pozadí byl k segmentaci obrazu použit jednoduchý model průměru pozadí získaný průměrováním pozadí ručně získaného v různé dny. Přestože se ukázalo, že tento způsob detekce prasete je pro detekci bílých prasat dostatečně přesný, potřebuje mít předem známý model pozadí. Pokud se však snímání kotec změní, model přestane fungovat.

Robustnějším přístupem je časté aktualizování modelu pozadí. Tím je také dosažena nejvyšší efektivita potlačení kolísání intenzity osvětlení v průběhu dne.

### 5.2.1 Mediánový model pozadí

Nejintuitivnějším přístupem je udržovat si v paměti posledních  $K$  snímků a odečítat aktuální snímek od kombinace těchto  $K$  snímků. Kombinací posledních snímků může být jejich průměr a nebo právě medián intenzit každého pixelu v jednotlivých kanálech. Zvolení mediánu je výhodné v tom, že se do jeho hodnoty méně promítají extrémní hodnoty.



**Obrázek 5.7.** Výsledky segmentace s využitím odečítání pozadí, obrázky v každém sloupci patří jednomu praseti. První řádek představuje použité modely pozadí, druhý řádek jsou aktuální zpracovávané snímky, na třetím řádku jsou zobrazeny odečtené snímky a na posledním jsou výsledné binární pole po prahování a odstranění malých regionů.

Protože výpočet mediánu se ale provádí v každém uvažovaném snímku, je tato metoda poměrně časově náročná. Problémem by také v některých aplikacích mohlo být udržovat si v paměti neustále posledních  $K$  snímků. V [13] (kap.16) je popsáno možné zrychlení výpočtu mediánového modelu pozadí. Model se vypočítá pouze na začátku videa z počátečních  $K$  snímků. Každý další snímek se do modelu promítne tak, že zvětší intenzitu jednotlivých pixelů o 1 (nebo jinou hodnotu), pokud je pixel v aktuálním snímku světlejší než pixel modelu, a nebo naopak zmenší hodnotu intenzity, pokud je aktuální snímek v daném pixelu tmavší než pixel modelu.

Výsledky zpracování videozámů prasat s využitím mediánového modelu pozadí jsou uvedeny až v následující sekci při porovnání s následující metodou.

### ■ 5.2.2 Pozadí jako směs gaussianů

Složitější metodou udržování a aktualizace pozadí je modelování pomocí směsi gaussianů. Tato metoda byla představena Staufferem a Grimsonem [15] a také je vysvětlena v knize Image Processing, Analysis and Machine Vision [13].

Základním postupem je přiřazení dat v každém snímku do několika gaussianů. Hledáme tedy  $K$  normálních (Gaussových) rozdělení pravděpodobnosti, která nejlépe reprezentují naše data. Každý pixel je ale modelován samostatně a má tak vlastní parametry jednotlivých gaussianů. Jednotlivé gaussiany se v čase aktualizují. Lze tedy předpokládat, že si takovýto model poradí s šumem i plynulými změnami osvětlení.

Vzhledem k tomu, že parametry jednotlivých rozdělení pro každý pixel videa jsou počítány v každém snímku, hodnota  $K$  se ve většině aplikací volí mezi 3 a 7 [13].

Jednotlivé gaussiany v čase  $t$  jsou definovány podle vztahu

$$N_{kt} = N(\mu_{kt}, \sigma_{kt}^2), \quad (5.13)$$

kde  $k = 1, \dots, K$ ,  $\mu$  a  $\sigma^2$  jsou parametry uvažovaného normálního rozdělení.

Každý gaussian má přiřazenou váhu  $\omega_{kt}$ , která je také aktualizována v čase. Tyto váhy jsou normalizovány a jejich součet je roven 1.

Pravděpodobnost pozorování pixelu s intenzitou  $g_t$  (v čase  $t$ ) je poté

$$P(g_t) = \sum_{k=1}^K \omega_{kt} \frac{1}{\sqrt{2\pi}} e^{-\frac{(g_t - \mu_{kt})^2}{\sigma_{kt}^2}}. \quad (5.14)$$

Při zpracovávání každého snímku musí být jednotlivé parametry ( $\mu$ ,  $\omega$ ,  $\sigma^2$ ) aktualizovány. Nejprve je hodnota intenzity uvažovaného pixelu porovnána s jednotlivými gaussiany pro daný pixel. Pokud se vyskytuje v prahem předem určeném okolí střední hodnoty uvažovaného gaussianu, je vybráno toto rozdělení. Pokud se žádné odpovídající rozdělení nenajde, je vytvořen nový gaussian. Protože gaussianů je stále udržováno  $K$ , k vytvoření nového musí být jeden odstraněn. Tím je gaussian s nejnižší vahou  $\omega$ .

Pokud je pixel zařazen do jednoho z gaussianů, jsou všechny váhy jednotlivých gaussianů, do kterých pixel přiřazen nebyl, přepočítány podle vztahu

$$\omega_{kt} = (1 - \alpha)\omega_{k(t-1)}, \quad (5.15)$$

kde  $\alpha$  je učicí konstanta. Tato konstanta udává, jak rychle se má model obnovovat. Váha gaussianu, do kterého byl aktuální pixel přiřazen, se ponechá stejná. Po tomto kroku musí být váhy opět normalizovány.

Parametry gaussianu, do kterého byl pixel přiřazen se upraví podle vztahů

$$\begin{aligned} \mu_{kt} &= (1 - \rho)\mu_{k(t-1)} + \rho g_t, \\ \sigma_{k(t-1)}^2 &= (1 - \rho)\sigma_{k(t-1)}^2 + \rho(g_t - \mu_{kt})^2, \end{aligned} \quad (5.16)$$

kde

$$\rho = \alpha P(g_t | \mu_k, \sigma_k^2).$$

Posledním úkolem je rozhodnout, které gaussiany jsou pozadím a která popředím – naším sledovaným objektem. Toto rozdělení je založeno na předpokladu, že pozadí se



mění velmi pomalu a tak gausiány pixelů odpovídajících pozadí budou mít stálé nebo pomalu se měnící parametry.

Gausiány jsou seřazeny podle podílu parametrů

$$\frac{\omega_{kt}}{\sigma_{kt}}$$

Vyšší hodnoty uvažovaného podílu patří rozdělením s malým rozptylem nebo vysokou vahou nebo oběma variantami. Pro pixely, kterým taková rozdělení patří, platí, že se v čase málo mění – jsou pozadím. Naopak pixely, které předešlé vlastnosti nesplňují, jsou klasifikovány jako popředí.

Pozadím jsou pro každý pixel gausiány  $k = 1, \dots, B$ , kde

$$B = \operatorname{argmin}_b \left( \sum_{k=1}^b \omega_{kt} > T \right) \quad (5.17)$$

a  $T$  je předem zvolený práh.

Pixel je klasifikován jako popředí, pokud nebyl zařazen do žádného již existujícího rozdělení pro daný pixel modelu nebo byl přidělen do takového rozdělení, které není pozadím.

### ■ 5.2.3 Implementace a diskuze výsledků

Ke zpracování videa byla použita implementace modelu pozadí jako směsi gausiánů podle Image Processing, Analysis, and Machine Vision - A MATLAB Companion [16].

Na následujících obrázcích č. 5.8 a č. 5.9 jsou vidět výsledky zpracování videí prasat uvedenými metodami. V pravém sloupci jsou vidět binární obrazy s klasifikací popředí(černé)/pozadí(bílé).

Uvedené metody sledování pohybu s využitím průběžného modelu pozadí ukázaly, že je s jejich pomocí možné úspěšně sledovat pohyb prasete. Malou nevýhodou je, že první snímky nejsou hlavně u gausiánového modelu pozadí vypovídající, protože model ještě nebyl plně inicializován a váhy jednotlivých rozdělení u jednotlivých pixelů ještě úplně neoddelují pozadí od popředí.

Výše popsané metody fungují velmi dobře, pokud se prase hýbe a nestojí na místě příliš dlouho. To je vidět zejména na obrázcích mediánového modelu pozadí ve 135. a 750. vteřině u obou obrázků. V obou ukázkových videích se začne prase pomalu promítat do modelu pozadí a při odečítání aktuálního snímku a pozadí poté prase není segmentováno celé, ale je vidět pouze jeho část, u které byl zaznamenán pohyb. Například pokud prase dlouho leží a pak začne hýbat hlavou, je možné, že se takový pohyb ztratí při filtrování výsledného obrazu od šumu a falešně pozitivních klasifikací pixelů popředí.

Jak je vidět na obr. č. 5.8, bílé prase je v obraze poměrně úspěšně segmentováno oběma metodami. Docela dobré výsledky jsou vidět také pro hnědé prase na obr. č. 5.9, zde je však více poznat šum a falešně pozitivní detekce popředí zejména u modelu pomocí směsi gausiánů.

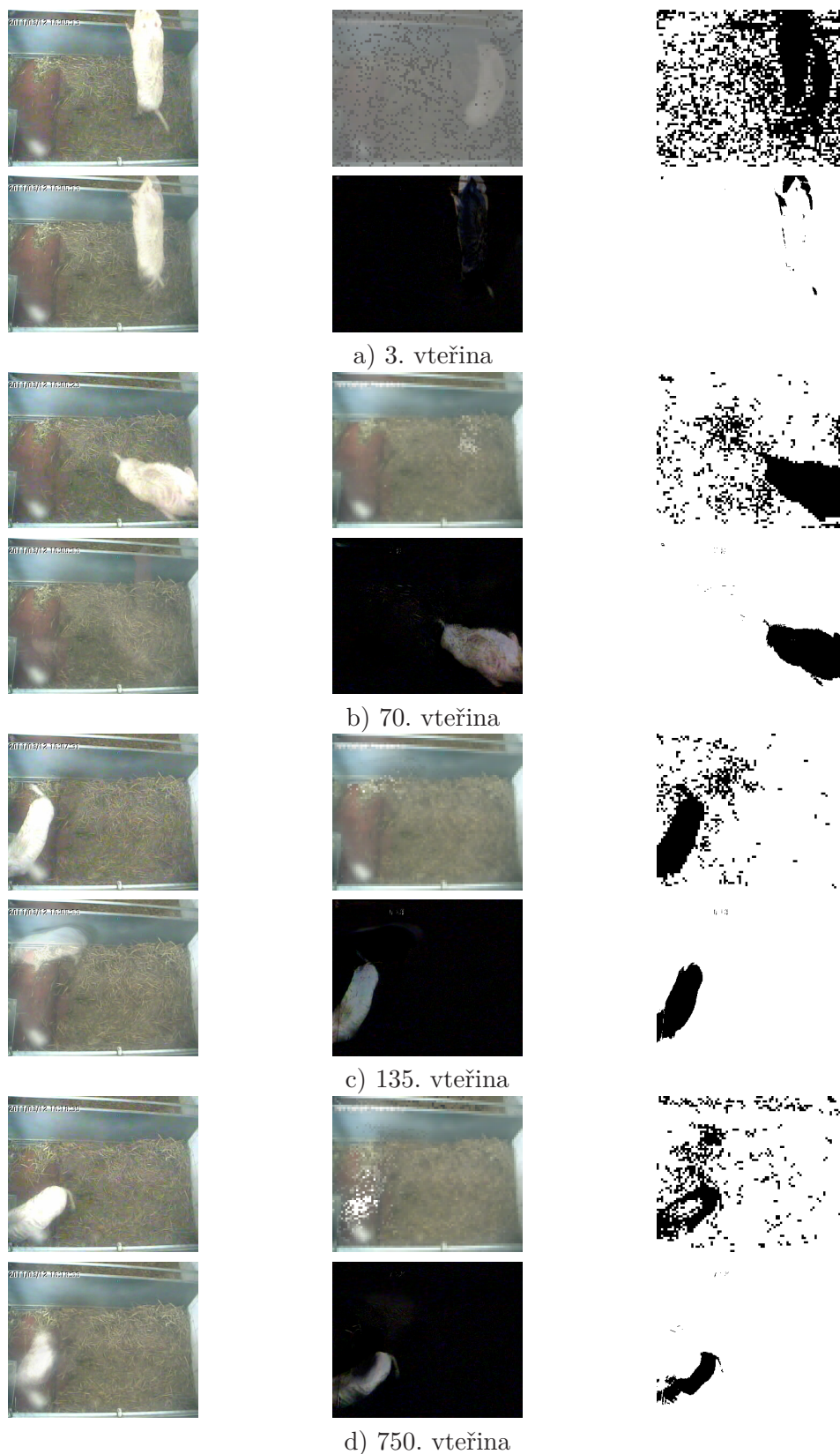
Lze tedy poměrně přesně určit stávající polohu prasete, oba modely však ztrácí přesný obrys zvířete, pokud se nějakou dobu nehýbe. Doba nečinnosti potřebná ke splnutí zvířete a pozadí je u gausiánové metody delší. I tak se však tato doba u obou modelů pohybuje v řádu desítek uvažovaných snímků.

I kvůli této vlastnosti bylo video zpracovávalo nikoliv s nahrávací frekvencí 24 snímků za vteřinu, ale pouze s frekvencí 1 snímek za vteřinu. Zpracovával se tedy pouze každý 24. snímek. Toto opatření pomohlo zmenšit vliv splývání zvířete s obrazem zejména v úsecích videa, kdy se prase nehýbe.

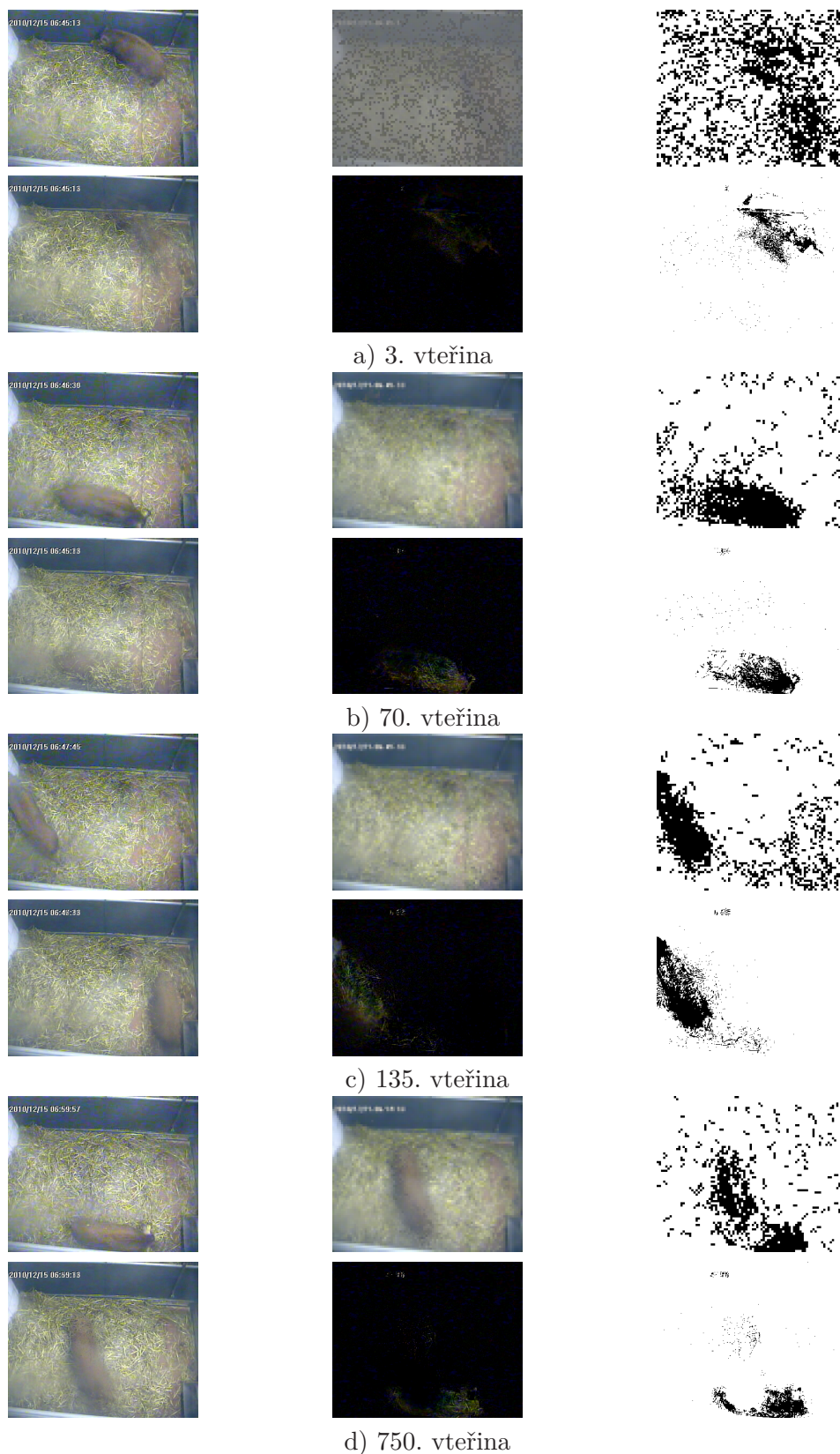
Použití těchto metod je tedy vhodné spíše v takových aplikacích, kdy sledované objekty nejsou stále přítomné v prostoru, který je nahráván. Jako příklad lze uvést městskou křižovatku, kterou auta projíždějí pouze s možností krátkého zastavení. Na tomto příkladu metodu využívající směs gausiánů prezentuje Stauffer a Grimson [15].

Zde je také ukázána velká výhoda obou výše popsaných metod pro modelování pozadí, kterou je přizpůsobení změnám osvětlení v průběhu dne.

Obě otestované metody lze z hlediska rychlosti použít pro sledování videa v reálném čase. Pro zpracování několika desítek hodin videí v co nejkratším čase je však použitelná pouze metoda s mediánovým modelem pozadí, která je řádově rychlejší. Druhá popsaná metoda zvládne za vteřinu zpracovat pouze jednotky snímků.



**Obrázek 5.8.** Výsledky zpracování videa metodami modelování pozadí. Obrázky zleva doprava odshora dolů: aktuální snímek, pozadí namodelováno pomocí směsi gaussiánů, detekované popředí pomocí směsi gaussiánů, mediánový model pozadí, rozdíl snímku a mediánového modelu pozadí, segmentovaný rozdíl snímků.



**Obrázek 5.9.** Výsledky zpracování videa metodami modelování pozadí. Obrázky zleva doprava odshora dolů: aktuální snímek, pozadí namodelováno pomocí směsi gaussiánů, detekované popředí pomocí směsi gaussiánů, mediánový model pozadí, rozdíl snímku a mediánového modelu pozadí, segmentovaný rozdíl snímků.



# Kapitola 6

## Sledování na základě optického toku

Optický tok je rozdělení zdánlivých rychlostí pohybu (nebo spíše změny barevné intenzity intenzity) v obraze. Pro výpočet je zapotřebí minálně dvou po sobě jdoucích snímků. Optický tok lze použít k segmentaci obrazu na základě mezisnímkové změny. Tato metoda se používá pro sledování pohybujícího se objektu, ale také pro zjištění pohybu kamery vzhledem k nehybným objektům.

Základní a velmi používaný přístup k výpočtu optického toku je představen Hornem a Schunckem v [17].

Horn a Schunck zavádějí ve svém článku intenzitu pixelu na pozici  $[x, y]$  v čase  $t$  jako  $E(x, y, t)$ . Dále je ukázáno, že platí rovnice

$$\frac{\partial E}{\partial x} \frac{dx}{dt} + \frac{\partial E}{\partial y} \frac{dy}{dt} + \frac{\partial E}{\partial t} = 0. \quad (6.1)$$

Dále jsou vyjádřeny jednotlivé rychlosti

$$u = \frac{dx}{dt}, \quad (6.2)$$

$$v = \frac{dy}{dt}. \quad (6.3)$$

Cílem je pro každý bod určit rychlosti  $u$  a  $v$ . Rovnici č. (6.1) lze přepsat do jednoduššího tvaru

$$E_x u + E_y v + E_t = 0, \quad (6.4)$$

kde  $E_x$ ,  $E_y$  a  $E_t$  jsou členy odpovídající jednotlivým parciálním derivacím. Tyto parciální derivace jsou v každém bodě  $[i, j]$  aproximovány podle následujících rovnic (v potaz jsou brány dva po sobě jdoucí snímky  $k$  a  $k + 1$ )

$$E_x \approx \frac{1}{4}(E(i, j + 1, k) - E(i, j, k) + E(i + 1, j + 1, k) - E(i + 1, j, k) + E(i, j + 1, k + 1) - E(i, j, k + 1) + E(i + 1, j + 1, k + 1) - E(i + 1, j, k + 1)), \quad (6.5)$$

$$E_y \approx \frac{1}{4}(E(i + 1, j, k) - E(i, j, k) + E(i + 1, j + 1, k) - E(i, j + 1, k) + E(i + 1, j, k + 1) - E(i, j, k + 1) + E(i + 1, j + 1, k + 1) - E(i, j + 1, k + 1)), \quad (6.6)$$

$$E_t \approx \frac{1}{4}(E(i, j, k + 1) - E(i, j, k) + E(i + 1, j, k + 1) - E(i + 1, j, k) + E(i, j + 1, k + 1) - E(i, j + 1, k) + E(i + 1, j + 1, k + 1) - E(i + 1, j + 1, k)). \quad (6.7)$$

Určení rychlostí  $u$  a  $v$  probíhá minimalizací součtu chyb

$$\epsilon^2 = \alpha^2 \epsilon_c^2 + \epsilon_b^2, \quad (6.8)$$

kde  $\epsilon_b = E_x u + E_y v + E_t$ ,  $\epsilon_c^2 = (\bar{u} - u)^2 + (\bar{v} - v)^2$ ,  $\bar{u}$  a  $\bar{v}$  jsou lokální průměry jednotlivých rychlostí a  $\alpha$  je váhová konstanta.

Výše popsaná minimalizace by však byla příliš výpočetně náročná a proto je předvedeno iterativní řešení. Lokální průměry zdánlivých rychlostí  $u$  a  $v$  jsou vypočítány podle vztahů

$$\begin{aligned} \bar{u}(i, j, k) &= \frac{1}{6}(u(i-1, j, k) + u(i, j+1, k) + u(i+1, j, k) + u(i, j-1, k)) \\ &+ \frac{1}{12}(u(i-1, j-1, k) + u(i-1, j+1, k) + u(i+1, j+1, k) + u(i+1, j-1, k)), \end{aligned} \quad (6.9)$$

$$\begin{aligned} \bar{v}(i, j, k) &= \frac{1}{6}(v(i-1, j, k) + v(i, j+1, k) + v(i+1, j, k) + v(i, j-1, k)) \\ &+ \frac{1}{12}(v(i-1, j-1, k) + v(i, j+1, k) + v(i+1, j+1, k) + v(i+1, j-1, k)). \end{aligned} \quad (6.10)$$

Nové rychlosti jsou vypočítány iterativně následujícím způsobem

$$u^{n+1} = \bar{u}^n - E_x[E_x \bar{u}^n + E_y \bar{v}^n + E_t]/(\alpha 2 + E_x^2 + E_y^2), \quad (6.11)$$

$$v^{n+1} = \bar{v}^n - E_y[E_x \bar{u}^n + E_y \bar{v}^n + E_t]/(\alpha 2 + E_x^2 + E_y^2). \quad (6.12)$$

Počet iterací je předem znám a je konstantní.

K analýze pohybu prasete byl použit kód, který vychází z odvození výše, implementovaný Jasonem Yu-Tseh Chim<sup>1)</sup>. Výsledky jsou vidět na obrázcích č. 6.1 a č. 6.2.

Nevýhodou této metody je její pomalost. I při iterativním počítání rychlostí  $u$  a  $v$  trvá zpracovat jeden snímek v prostředí MATLAB od 0,5 do 3 vteřin. Přesný čas záleží na počtu zvolených iterací při výpočtu rychlostí. Další nevýhodou je rozpoznání pouze obrysu prasete. To je způsobeno homogenitou hřebu prasete.

Metoda sledování prasete pomocí optických toků ale dokáže lokalizovat i pohybující se hnědé prase. U všech tří prasat není rozlišena hlava s dostatečnou přesností. To je způsobeno tím, že barva hlavy a části boku prasat splývá s podestýlkou.

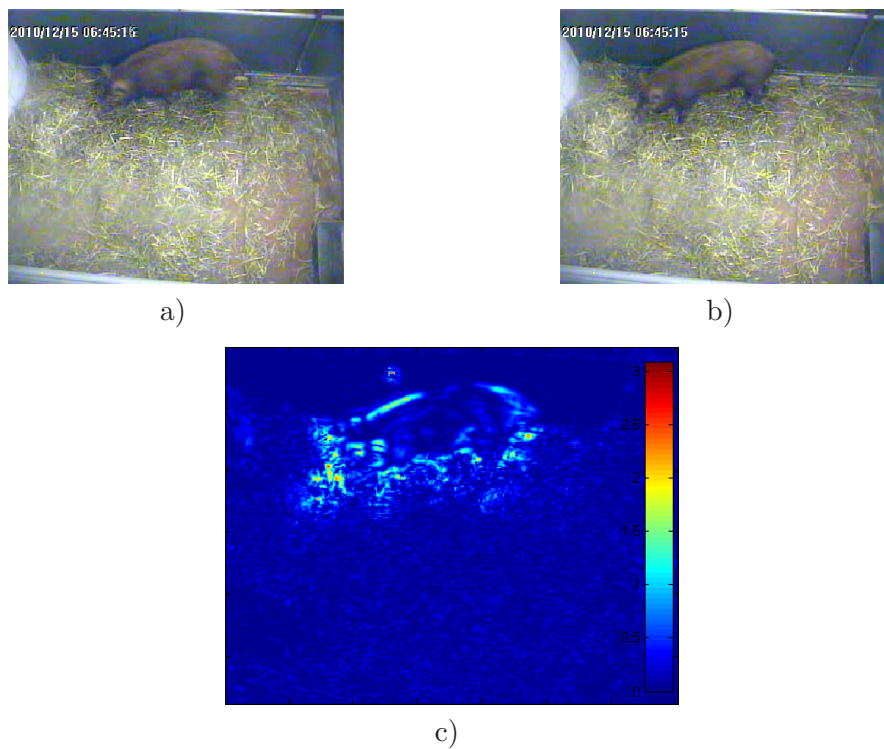
Velmi přesný algoritmus pro určení optického toku představuje Liu ve své doktorské práci [18]. Ten se zakládá na mnoha pokročilých metodách. K výše představené Hornově-Schunckově metodě používá dále Lucasovu-Kanadeho metodu a implementuje řadu matematických transformací a rychlou metodu řešení diferenciálních rovnic. Více informací a samotný kód v prostředí MATLAB je k dispozici zde<sup>2)</sup>.

Výsledky Liuovy metody jsou vidět na obrázcích č. 6.3 a č. 6.4. Je vidět, že prase je i při malém pohybu velmi úspěšně detekováno. To platí pro obě prasata, tedy pro bílé i pro hnědé. Jeden rozdíl dvou snímků trvá na průměrném PC zpracovat přibližně 1 vteřinu.

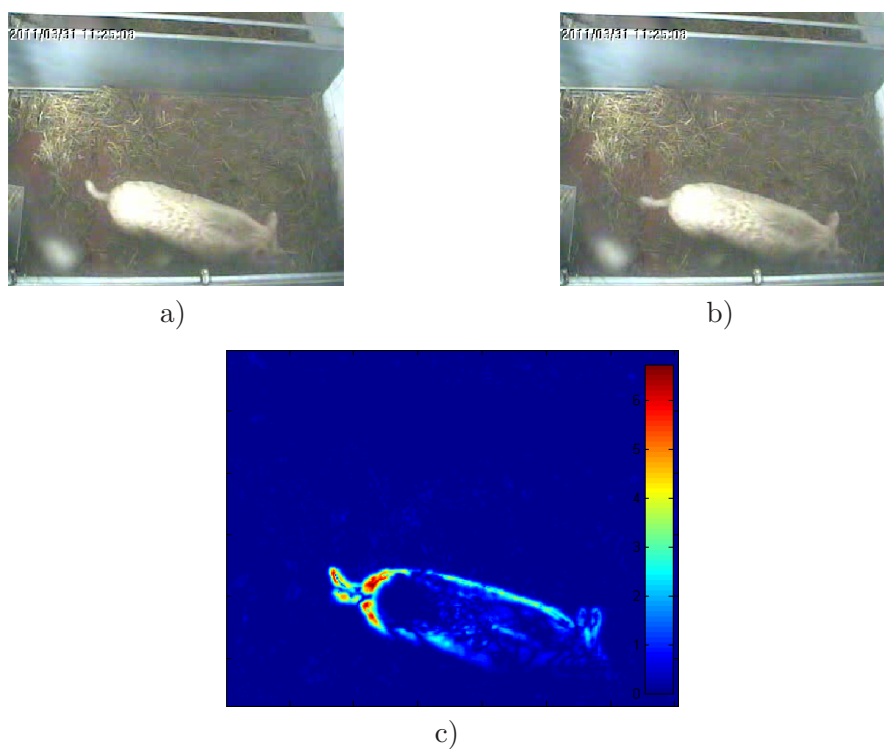
I když vzhledem k časové náročnosti není možné zpracovávat videa v reálném čase, pro pokročilou analýzu vybraných částí dne by tato metoda mohla velmi dobře posloužit.

<sup>1)</sup> <http://chi3x10.wordpress.com/2008/06/09/optical-flow-in-matlab/>

<sup>2)</sup> <http://people.csail.mit.edu/celiu/OpticalFlow/>

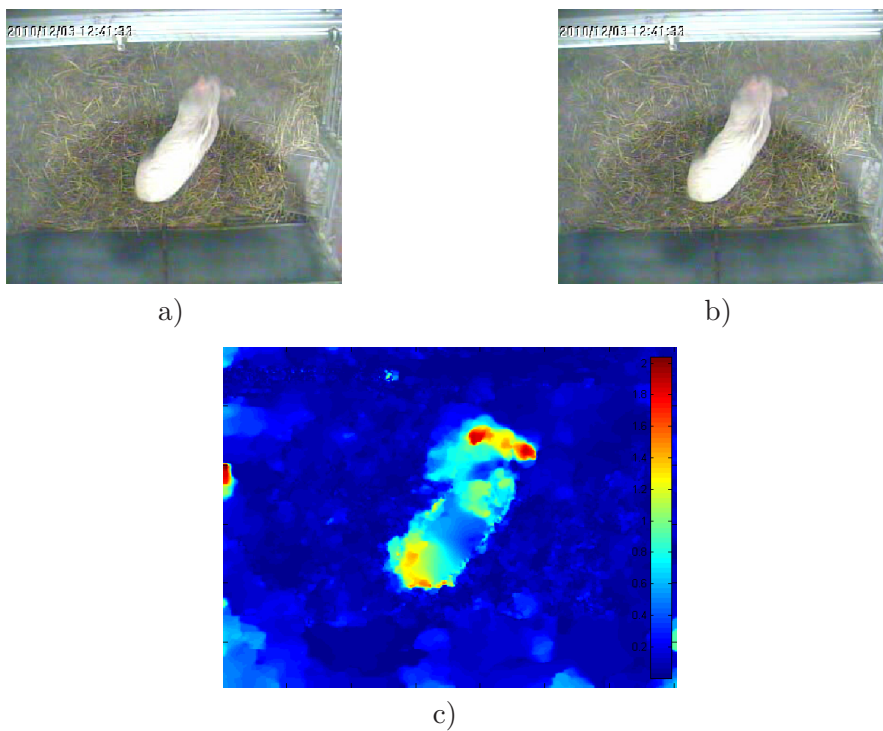


**Obrázek 6.1.** Výsledky sledování prasete Bruna pomocí optických toků. a) první snímek sekvence, b) druhý snímek sekvence, c) amplituda vektoru rychlosti  $\mathbf{c} = (u, v)$ .

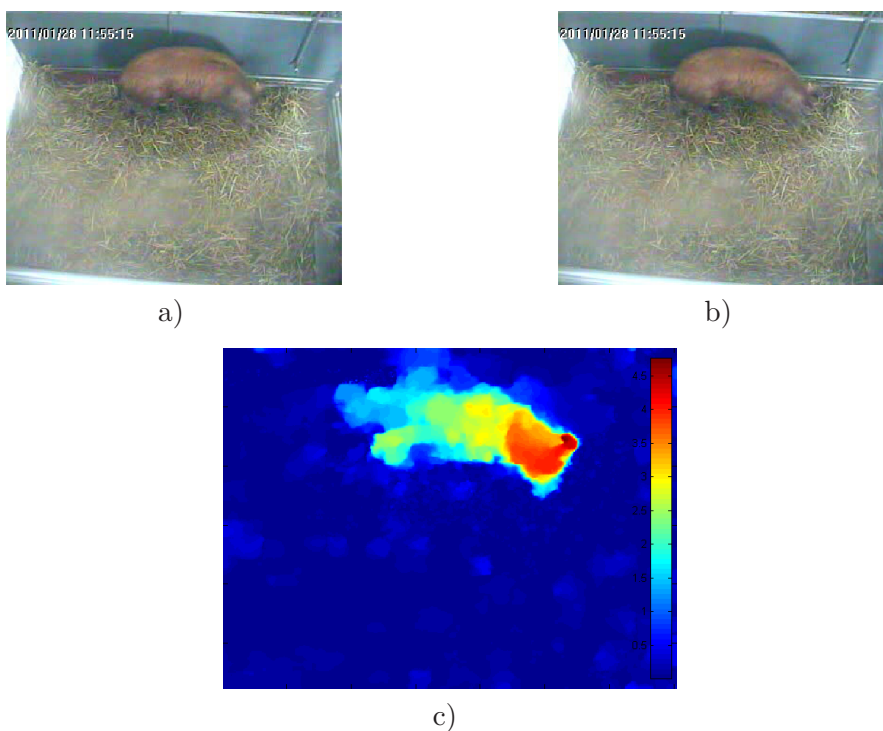


**Obrázek 6.2.** Výsledky sledování prasete Alfa pomocí optických toků. a) 1. snímek sekvence, b) druhý snímek sekvence, c) amplituda vektoru rychlosti  $\mathbf{c} = (u, v)$ .





**Obrázek 6.3.** Výsledky sledování prasete Babe pomocí optických toků. a) první snímek sekvence, b) druhý snímek sekvence, c) amplituda vektoru rychlosti  $\mathbf{c} = (u, v)$ .



**Obrázek 6.4.** Výsledky sledování prasete Bruna pomocí optických toků. a) první snímek sekvence, b) druhý snímek sekvence, c) amplituda vektoru rychlosti  $\mathbf{c} = (u, v)$ .

## Kapitola 7

### Kernel-based sledování

Další známý typ sledování pohybujícího se objektu ve scéně je tzv. kernel-based sledování. Tato metoda je založena na použití prostorové masky, kterou je označen objekt, a ta se poté posouvá obrazem stejně tak, jak se sledovaný objekt hýbe. Také je využíváno předpokladu, že pozice objektu a také objekt sám se mezi jednotlivými snímky příliš nemění. Metoda byla představena Comanicu a kol. v publikacích [11] a [19]. Jako masku používá Comanicu a kol. elipsu, což je výhodné i pro naši aplikaci.

V samotném sledování je zapotřebí dvou základních struktur – cílového modelu a kandidátů. Tyto struktury jsou reprezentovány hustotou pravděpodobnosti výskytu předem zvolených parametrů jednotlivých pixelů. Často je používáno postačující modelování pomocí  $m$ -intervalového tzv.  $m$ -binového histogramu [13] (str. 480-486).

Rozdělení pixelů do jednotlivých binů je ještě doplněno váhováním jednotlivých pixelů podle vzdálenosti od středu uvažované masky. K tomu jsou využívány speciální funkce tzv. kernelové funkce.

Na začátku videozáznamu je zvolena maska, v našem případě elipsa, ve které je vypočítán model  $\mathbf{q}_u$ . Výpočet se provede podle následujícího vztahu:

$$\mathbf{q}_u = C \sum_{i=1}^n k(\|\mathbf{x}_i\|^2) \delta(b(\mathbf{x}_i^*) - u), \quad (7.1)$$

kde  $\mathbf{x}^*$  jsou jednotlivé pixely normalizované vzhledem ke středu prostorové masky,  $n$  je počet pixelů masky,  $\delta$  je funkce Kroneckerovo delta,  $k$  je kernelová funkce,  $b$  je funkce přiřazující hodnoty intenzit pixelů do jednotlivých binů histogramu ( $m$  binů),  $u$  je označení pro konkrétní interval histogramu (bin) a  $C$  je normalizační konstanta

$$C = \frac{1}{\sum_{i=1}^n k(\|\mathbf{x}_i\|^2)}. \quad (7.2)$$

Pro výpočet pohybu objektu a tedy i pohybu uvažované masky se vypočítá model prvního cílového kandidáta  $\mathbf{p}_u(\mathbf{y})$  podle

$$\mathbf{p}_u(\mathbf{y}) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{\mathbf{y} - \mathbf{x}_i}{h}\right\|^2\right) \delta(b(\mathbf{x}_i^*) - u), \quad (7.3)$$

kde  $\mathbf{y}$  je střed aktuální uvažované masky,  $h$  je její velikost (parametrem  $h$  se normuje vzdálenost jednotlivých pixelů masky od jejího středu) a  $C_h$  je opět normalizační konstanta

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k\left(\left\|\frac{\mathbf{y} - \mathbf{x}_i}{h}\right\|^2\right)}. \quad (7.4)$$

Normalizační konstanty  $C$  a  $C_h$  se používají pro zajištění toho, aby sumy přes všechny prvky modelu a kandidáta byly rovny 1.

K výpočtu nové polohy masky je také potřeba umět posoudit rozdílnost modelu od kandidáta. K tomuto se používá tzv. Bhattacharyya koeficient dán vztahem

$$\rho(\mathbf{y}) \equiv \rho[\mathbf{p}(\mathbf{y}), \mathbf{q}] = \sum_{i=1}^m \sqrt{p_u(\mathbf{y})q_u}. \quad (7.5)$$

Bhattacharyya koeficient je v tuto chvíli vypočítán pro vzdálenost mezi modelem a kandidátem se stejným středem jako střed modelu (s označením  $\rho[\mathbf{p}(\mathbf{y}_0), \mathbf{q}]$ ). Účelem popisovaného sledovacího algoritmu je tuto podobnost maximalizovat.

Za pomoci vypočítaného cílového modelu  $\mathbf{q}_u$  a modelu kandidáta  $\mathbf{p}_u(\mathbf{y})$  je pro každý pixel masky vypočítána váha  $\mathbf{w}$  podle vztahu

$$w_i = \sum_{i=1}^m \sqrt{\frac{\mathbf{q}_u}{\mathbf{p}_u(\mathbf{y})}} \delta(\mathbf{b}(\mathbf{x}_i^*) - u). \quad (7.6)$$

Nová pozice středu masky  $\mathbf{y}_1$  je vypočítána s použitím vah  $\mathbf{w}$  takto

$$\mathbf{y}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i g(\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\|^2)}{\sum_{i=1}^{n_h} \mathbf{x}_i g(\|\frac{\mathbf{y}-\mathbf{x}_i}{h}\|^2)}. \quad (7.7)$$

Kernel-based sledování je sledováním iterativním. Po vypočítání nového středu masky  $\mathbf{y}_1$  je vypočítána vzdálenost  $\rho[\mathbf{p}(\mathbf{y}_1), \mathbf{q}]$  takového kandidáta od modelu.

Na základě rozdílu mezi  $\rho[\mathbf{p}(\mathbf{y}_0), \mathbf{q}]$  a  $\rho[\mathbf{p}(\mathbf{y}_1), \mathbf{q}]$  se iteruje k nalezení nového středu  $\mathbf{y}_1$ . Poté se přejde znovu k přepočítání vah a novému výpočtu jednotlivých Bhattacharyya koeficientů, dokud není rozdíl  $\mathbf{y}_0$  a  $\mathbf{y}_1$  menší než předem zvolený práh. Podrobný popis algoritmu je k dispozici v [13] (str. 783) a [19].

Podle [13] (str. 480-486), [19] a [16] (str. 242-247) byl sledovací algoritmus naimplementován a odzkoušen na dostupných videozáznamech.

Kód dostupný v [16] je velmi rychlý a video dokáže sledovat v reálném čase i v pomalém prostředí MATLAB. To je dáno také tím, že sledovanou maskou je kruh. Podle [16] a [19] bylo implementováno také sledování eliptické masky. Výpočet je ale pomalejší než při sledování jednoduššího tvaru z důvodu neustálého přepočítávání vnitřních pixelů masky, které je u elipsy složitější a pomalejší.

Představený kód ale nepočítá s rotací sledovaného objektu. Úpravu pro rychlé sledování rotace přináší Zhang a kol. [12]. Tato modifikace byla implementována, ale ukázalo se, že představená metoda není příliš vhodná kvůli ještě složitějšímu udržování vnitřních bodů masky a bodů modelu.

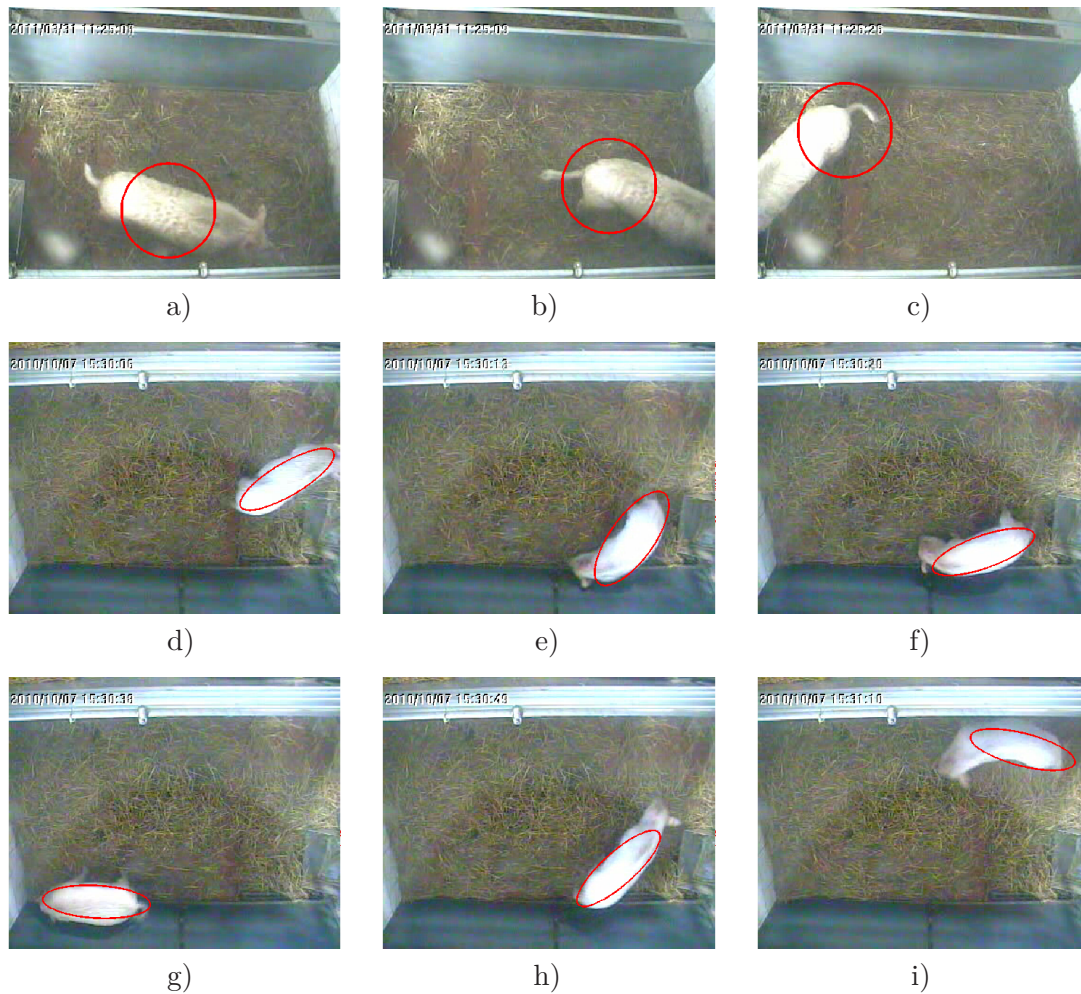
Namísto toho bylo použito jednoduchého přístupu výpočtu podobnosti mezi aktuální pozicí a rotací masky a modelem. Takováto podobnost se vypočítala pro malé rozmezí úhlů natočení elipsově masky a byl vybrán takový úhel, ke kterému patří nejvyšší podobnost.

Výsledky představených metod jsou zobrazeny na obr. č. 7.1. Pozice počítačnické masky (kruhu a elipsy) byly umístěny ručně.

Z výsledků je vidět, že pro sledování protáhlého tvaru prasete, které je navíc na svém hřebě poměrně homogenní, co se týče barvy a struktury, je výhodnější použít eliptickou masku. Kruhovat totiž právě kvůli homogenitě z prasete sjede až k jeho kraji.

Algoritmus sledování prasete s pomocí eliptické masky a jednoduchého sledování úhlu natočení má poměrně velkou přesnost, jak je vidět na případu prasete Babe (spodní řada) na obr. č. 7.1. Rychlost výpočtu v prostředí MATLAB je pro metodu s výpočtem úhlu natočení přibližně 30 snímků za vteřinu. Taková frekvence zpracovávání snímků stačí pro sledování v reálném čase.

Velkým problémem je však nerovnoměrné osvětlení kotce. V některých částech kotce je světelná intenzita větší než v jiných částech. Tím dochází k velkému rozdílu mezi



**Obrázek 7.1.** Výsledky Kernel-based sledování prasete v obraze. V první řadě je sledování prasete Alf pomocí rychlejšího algoritmu využívající jednodušší masku, v druhé a třetí řadě je zobrazeno prase Babe s eliptickou maskou.

sledovaným objektem a modelem a může se stát, že maska uvízne v bodech se stálou vysokou intenzitou, jako je například zeď.

## Kapitola 8

# Metody založené na geometrickém modelu popředí

Všechny představené metody poskytly jako výstup množinu bodů klasifikovanou buď jako celé prase nebo jako jeho část, která je v pohybu. Z těchto bodů se dá pouze pomocí nalezení jejich středu zjistit translace prasete mezi jednotlivými snímky. Další analýza je vhodná k použití zejména pro segmentační metody sledování zvířete v obrazu, protože jsou časově i implementačně méně náročné než modelování pozadí jako směs gaussianů nebo kernel-based sledování, a proto k jejich využití zbývá více výpočetního času.

Další velkou výhodou, proč k analýze použít jednodušší metody, je segmentace velké části prasete a nikoli pouze jeho pohybující se části.

Z pohybu prasete by se však dalo vyčíst o mnoho více než pouze translace jeho středu. Pokud by se prase točilo na místě, jeho translační pohyb by byl zaznamenán jako nulový.

Metodami, kterými lze z klasifikovaných bodů vyčíst další parametry, se zabývá tato kapitola.

### 8.1 Proložení elipsou

Prvním parametrem, který by bylo vhodné k translaci prasete doplnit, je jeho rotace. To se dá poměrně jednoduše zajistit tím, že se klasifikovaná data proloží elipsou, ze které se úhel natočení dá snadno vypočítat. Čím přesněji je prase segmentováno a čím větší je správně klasifikovaná množina bodů, tím přesnější bude výpočet natočení prasete s využitím proložení dat elipsou.

Rychlou, přesnou a spolehlivou metodu k proložení dat elipsou nabízí Fitzgibbon a kol. [20]. Ve zmíněném článku je prezentována metoda, která k proložení nepotřebuje více než jednu iteraci.

Pro popis elipsy je využita funkce pro obecnou kuželosečku

$$F(\mathbf{a}, \mathbf{x}) = \mathbf{a} \cdot \mathbf{x} = ax^2 + bxy + cy^2 + dx + ey + f = 0, \quad (8.1)$$

kde  $\mathbf{a}$  je vektor parametrů  $\mathbf{a} = [a \ b \ c \ d \ e \ f]^T$  a  $\mathbf{x} = [x^2 \ xy \ y^2 \ x \ y \ 1]^T$  je vektor jednotlivých souřadnic.

Aby takto popsaná kuželosečka byla elipsou, musí parametry splňovat podmínku

$$4ac - b^2 = 1. \quad (8.2)$$

Tuto podmínku lze vyjádřit maticí

$$C = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (8.3)$$

A popis elipsy lze také zapsat maticí

$$\mathbf{D} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_n]^T, \quad (8.4)$$

kde  $n$  je počet uvažovaných bodů. Fitzgibbon ve svém článku ukazuje, že pro nalezení parametrického vektoru  $\mathbf{a}$  stačí vyřešit systém

$$\mathbf{S}\mathbf{a} = \lambda\mathbf{C}\mathbf{a}, \quad (8.5)$$

$$\mathbf{a}^T\mathbf{C}\mathbf{a} = 1, \quad (8.6)$$

kde  $S = \mathbf{D}^T\mathbf{D}$ , nalezením vlastních čísel  $\lambda_i$  a vlastních vektorů  $\mathbf{u}_i$ . Jediné řešení dává takový pár  $(\lambda_i, \mathbf{u}_i)$ , který má záporné vlastní číslo, hledaný vektor parametrů  $\mathbf{a}$  je poté vlastní vektor příslušný nalezenému zápornému vlastnímu číslu.

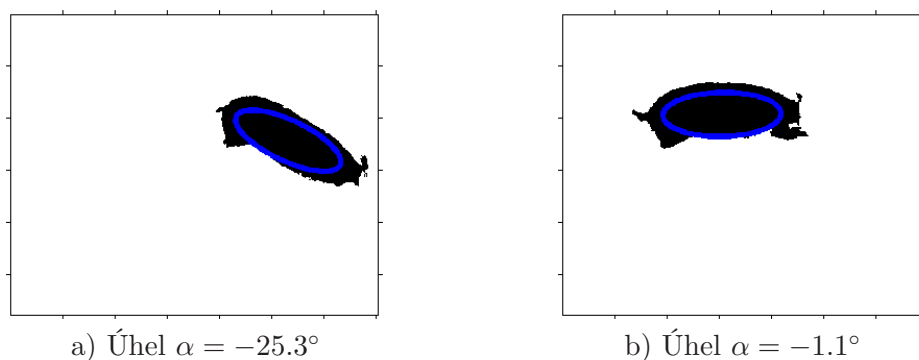
Úhel natočení takto popsané elipsy (mezi hlavní poloosou a osou  $x$  obrazu) se vypočítá podle vzorce

$$\alpha = \frac{1}{2} \tan^{-1}\left(\frac{b}{a-c}\right). \quad (8.7)$$

Při takovémto výpočtu se však musí brát v úvahu, že výsledný úhel je pouze z intervalu  $\langle -45^\circ, 45^\circ \rangle$  a přechod mezi krajními hodnotami tohoto intervalu musí být zaznamenán a obsažen ve výpočtu.

Výsledky proložení segmentovaných dat elipsou jsou vidět na obrázku č. 8.1. Proložení je přesné a odpovídá pozici a rotaci prasete. Při využití rychlé segmentace pomocí Otsuovy metody je s použitím prokládání elipsou pětiminutové video s nahrávací frekvencí 24 snímků za vteřinu (celkem 7200 snímků) zpracováno v prostředí MATLAB na počítači s procesorem Intel i5-2410M 2.3 Ghz při standardním běhu systému Windows 7 za 1 minutu.

Takovým algoritmem lze video analyzovat v reálném čase. Při hromadném zpracování pak lze nastavit výpočet elipsy pouze jednou za více snímků a tím zpracovávání úměrně urychlit.



**Obrázek 8.1.** Výsledky proložení segmentovaných dat odpovídajících praseti elipsou spolu s vypočteným úhlem.

## 8.2 Bodový model

Pohyb prasete byl zatím popsán jako translace a rotace. Pro sledování denních rytmů, tedy jestli prase leží či stojí (a kde se tomu tak děje) a nebo chodí (a jak rychle), jsou tyto dva ukazatele dostačující. Pro detailnější analýzu pohybů je však zapotřebí složitější

model. Jako příklad možného využití dalšího parametru je , že jednou z domněnek účinku Huntingtonovy choroby na zvířata je změna přímosti chůze.

Ke složitějšímu modelování prasete byl použit bodový model tvaru prasete, v literatuře dohledatelný pod anglickými názvy point-distribution model (PDM) nebo active shape model (ASM). Matematickým popisem získání modelu a jeho tréninku se zabývá Cootes a kol. [10], použití je také vysvětleno v knize [13] (str. 464-474) nebo příručce [16] (str. 144-153).

Základní myšlenkou modelu tvaru analyzovaného objektu je nalezení určitého průměrného tvaru, který uvažovaný objekt zaujímá, a poté přiblížení tohoto průměrnému tvaru tvaru aktuálnímu, ve kterém se sledovaný objekt právě nachází. Průměrný tvar se získá ručně z několika různých snímků. Nalezení takové transformace, kterou přejdeme z průměrného tvaru do tvaru aktuálního, lze dělat změnou parametrů uvažovaného bodového modelu nebo „přichycením“ jednotlivých bodů na obrys sledovaného objektu za dodržení specifických podmínek.

Prvním krokem při vytváření bodového modelu PDM je vhodnou transformací zarovnat trénovací set dat tak, aby se v budoucím porovnávání s dalšími sety dat opravdu porovnávalo podobné s podobným. Transformace se provádí tak, aby byl minimalizován součet kvadrátů vzdáleností (ve smyslu rozdílnosti modelů) podle vztahu

$$E = \sum_{i=1}^N w_i \left\| s \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} - \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} \right\|^2, \quad (8.8)$$

kde  $N$  je celkový počet bodů,  $w_i$  jsou váhy,  $s$  je parametr zvětšení/zmenšení,  $\theta$  je úhel natočení celého modelu,  $t_x$  a  $t_y$  jsou translace v jednotlivých osách,  $x_i$  a  $y_i$  jsou souřadnice jednotlivých bodů uvažovaného modelu a  $x'_i$  a  $y'_i$  jsou souřadnice bodů ostatních modelů.

Z takto vypočítaných transformovaných bodových modelů

$$\hat{\mathbf{x}}^i = [\hat{x}_1^i \quad \hat{y}_1^i \quad \hat{x}_2^i \quad \hat{y}_2^i \quad \dots \quad \hat{x}_M^i \quad \hat{y}_M^i],$$

kde  $M$  je počet bodových modelů, se průměrný tvar

$$\bar{\mathbf{x}}^i = [\bar{x}_1^i \quad \bar{y}_1^i \quad \bar{x}_2^i \quad \bar{y}_2^i \quad \dots \quad \bar{x}_M^i \quad \bar{y}_M^i]$$

spočítá jako aritmetický průměr jednotlivých souřadnic transformovaných bodových modelů.

Se znalostí průměru vypočítáme odchylku od něj

$$\delta \mathbf{x}^i = \hat{\mathbf{x}}^i - \bar{\mathbf{x}}^i. \quad (8.9)$$

Z této odchylky lze vypočítat kovarienční matici  $\mathbf{S}$  dimenze  $2N \times 2N$

$$\mathbf{S} = \frac{1}{M} \sum_{i=1}^M \delta \mathbf{x}^i (\delta \mathbf{x}^i)^T, \quad (8.10)$$

jejíž vlastní vektory  $\mathbf{p}^i$  jsou bází  $\mathbf{P} = [\mathbf{p}^1 \quad \mathbf{p}^2 \quad \dots \quad \mathbf{p}^{2N}]$ , kterou lze generovat jakýkoliv vektor bodového modelu  $\mathbf{x}$  pomocí vztahu

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b}, \quad (8.11)$$

kde  $\mathbf{b}$  je vektor odchylky bodového modelu od průměru vzhledem k danému vektoru báze.

Pokud ze snímku, ke kterému se snažíme přiřadit aktuální bodový model, dokážeme vyjádřit (segmentovat) prase s dostatečně přesnými hranami, lze na prase model nastavit iterativně. Požadavkem na funkčnost iterativního nastavení modelu je dostatečný počáteční odhad pozice, rotace a velikosti prasete.

Samotný algoritmus začíná výpočtem normálového vektoru ke křivce vedené body modelu v každém jednom bodě modelu. V obou směrech této normály do maximální vzdálenosti (dané předem definovaným prahem) od samotného bodu je zjištěn pixel největšího gradientu intenzity a ten je zvolen jako nová pozice bodu. Pokud na normále není žádný znatelný gradient, bod je zachován tam, kde byl. Takto je získána nová odchylka  $\tilde{\mathbf{x}}$  a nový model je přepočítán podle

$$\tilde{\mathbf{x}} + \delta\tilde{\mathbf{x}} = \tilde{\mathbf{x}} + \mathbf{P}(\mathbf{b} + \delta\mathbf{b}), \quad (8.12)$$

kde  $\delta\mathbf{b} = \mathbf{P}^T \delta\tilde{\mathbf{x}}$ .

Po přepočítání nového modelu se iteruje znovu od nalezení normál k jednotlivým bodům aktuálního modelu dokud není získána dostatečná přesnost. S takto vypočítaným modelem se případně pokračuje do dalšího snímku. Na obr. č. 8.2 je zobrazeno, jakým způsobem tento iterační algoritmus funguje. Segmentace je pro ukázkou získána pomocí Otsuovy metody a zadání středu prasete i jeho rotace bylo provedeno ručně.

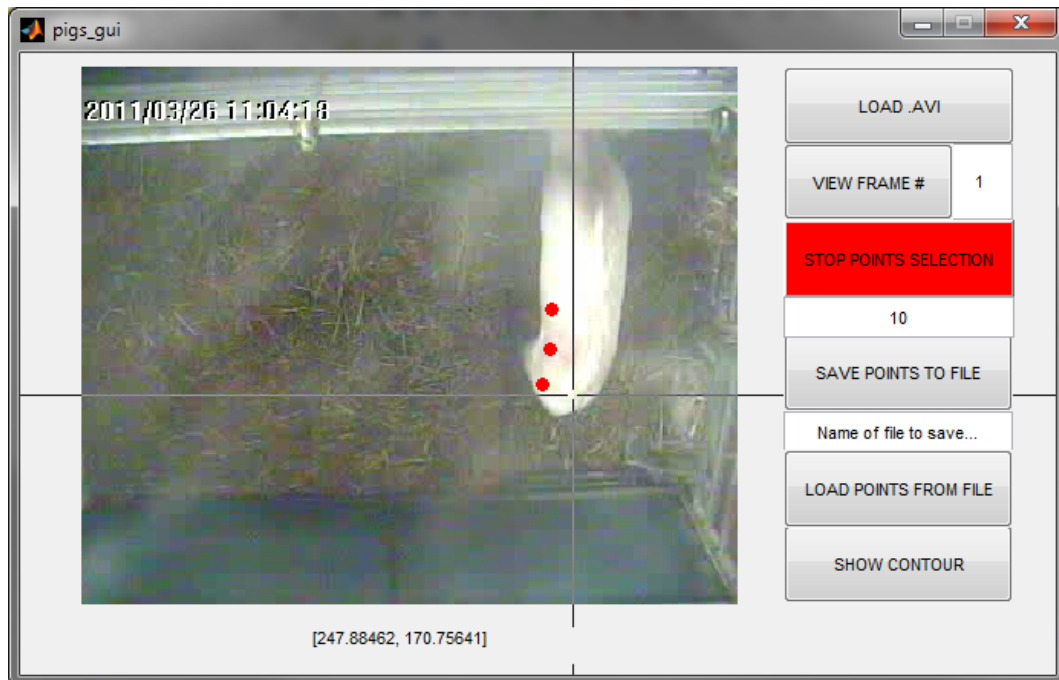
Pro zarovnání bodových modelů i pro jeho nastavení na aktuální tvar prasete byl použit kód v prostředí MATLAB z [16].



**Obrázek 8.2.** Ukázková práce algoritmu nastavení modelu na aktuální tvar prasete. Zelené úsečky jsou normály k tvaru prasete v uvažovaných bodech.

Pro získání modelu bylo navrženo a naprogramováno jednoduché uživatelské rozhraní do prostředí MATLAB. Toto rozhraní umožňuje rychlé a přehledné načtení požadovaného snímku a zadání jednotlivých bodů PDM. Tyto body lze poté uložit jako *\*.mat* soubor, ze kterého je možné body do prostředí opět načíst a kontrolovat je. Grafické uživatelské rozhraní je vidět na obr. č. 8.3.





**Obrázek 8.3.** Grafické uživatelské rozhraní pro získání bodového modelu PDM. Tlačítka po řadě “Nahrání .avi videa”, “Zobrazení konkrétního snímku (číslo ve vedlejším poli)”, “Začátek/konec zadávání bodů (počet bodů je zadáván o pole níže)”, “Ukládání bodů do souboru (název souboru o pole níže)”, “Nahrávání bodů ze souboru (název souboru o pole výše)”, “Proložení bodů křivkami”.



**Obrázek 8.4.** Ukázka ručně zadaných bodových modelů.

Ručně zadávané bodové modely 20 bodů jsou vidět na obrázku č. 8.4.

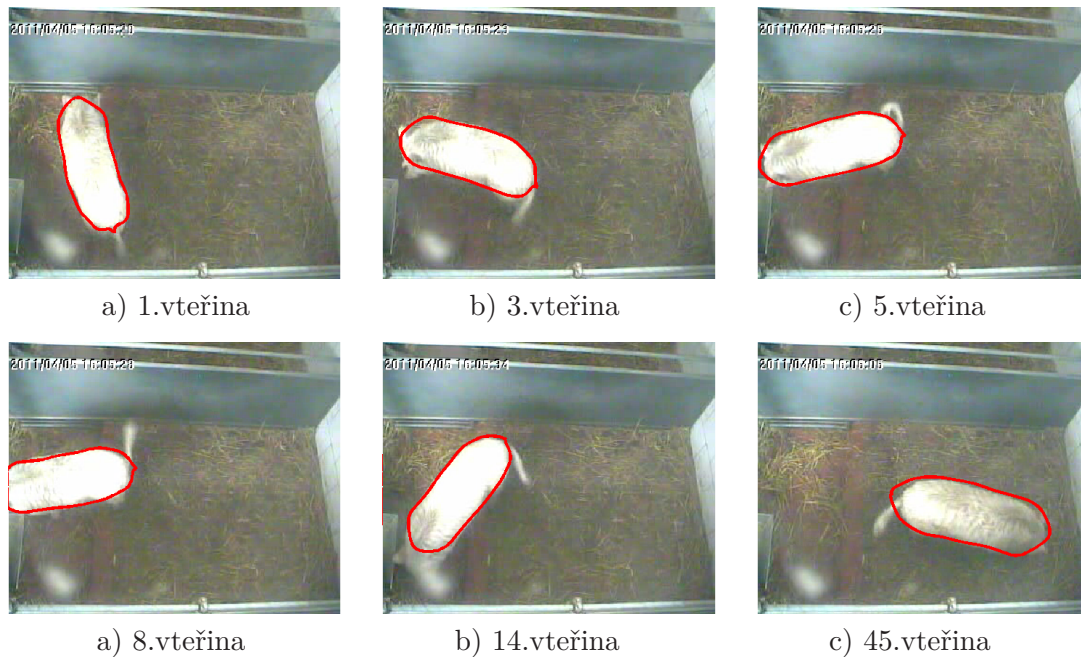
Marchant v [8] používá k modelování prasete 49 bodů. V našem případě bylo použito 10,14 a 20 bodů. Problémem stávajícího nahrávání prasat je však ten, že kamera není nastavena přímo nad středem kotce a nemíří na kotec z úhlu  $90^\circ$  a proto nejsou prasata snímána stále jen shora. Model se tedy bude zásadně lišit pro prase viděné téměř z boku a pro prase viděné shora.

Řešení tohoto problému v našem případě spočívalo v získání bodového modelu, který byl symetrický. Odpadla tak nutnost klasifikovat pozici hlavy prasete. Takový symetrický model se poté nastaví na aktuální pozici a tvar zvířete automaticky. Model si poradí i se situací, kdy je prase svou částí mimo obraz.

Výsledky při sledování bílého prasete jsou vidět na obrázku č. 8.5. Nastavení modelu na současnou polohu prasete je poměrně rychlé. Za vteřinu se pro dvacetibodový bo-

dový model stihne zpracovat 15 snímků. S omezením nahrávací frekvence lze tedy tuto metodu bez problémů použít pro sledování v reálném čase.

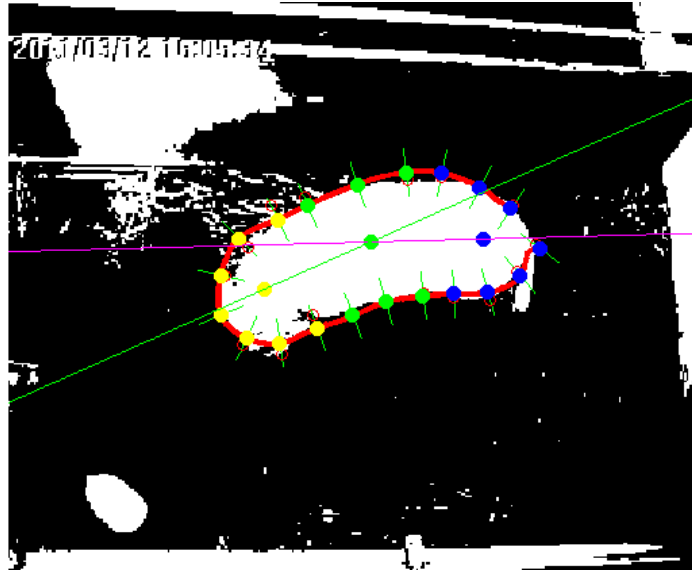
Jak je vidět z obrázku č. 8.5, sledování zvířete pomocí bodového modelu je velmi přesné a prase je spolehlivě reprezentováno. Nevýhodou této metody však je to, že velmi závisí na metodě segmentace, která dodává buď celé segmentované prase a nebo jeho obrys.



**Obrázek 8.5.** Sledování prasete s pomocí segmentace a bodového modelu.

Podnětem k využití a implementaci bodového modelu byla možnost vyčtení dalších parametrů. Původním plánem bylo získat z bodového modelu prohnutí prasete. Byla naprogramována metoda na získání středů bodů hlavy a zadní části prasete a tyto body spojit přímkou se středem bodů těla prasete. Prohnutím prasete se poté rozumí vzájemný úhel těchto přímk, ale také úhel s osami X a Y. Ukázalo se, že i takováto jednoduchá metoda je schopná dodat kvalitní výsledky. Příklad je k dispozici na obr. č. 8.6.

Nakonec ale bylo od další analýzy tímto způsobem upuštěno. Pozice kamery není pro takovéto získávání dat vhodná kvůli umístění moc nízko nad povrchem. Dále je v současné době stále problém se segmentací celé hlavy všech prasat, protože moc splývá s pozadím. Pro případné použití později však již jsou připraveny spouštěcí metody.



**Obrázek 8.6.** Příklad další analýzy aktuálního bodového modelu. Žluté body - zadní část prasete, zelené body - tělo prasete, modré body - hlava prasete.

# Kapitola 9

## Porovnání metod

Metody představené v minulých kapitolách byly pomocí implementovaného grafického uživatelského rozhraní (popsaného v kapitole Hromadné zpracování videa) otestovány na dostupných datech. Jejich průběh a výsledky jsou v této kapitole porovnány. Test proběhl na běžném stolním počítači se čtyřjádrovým procesorem Intel Core i5-2500K s taktovací frekvencí 3.3GHz a s operačním systémem Windows 7. Při testování bylo k uvažovanému PC přistupováno přes vzdálenou plochu. Měření rychlosti metod probíhalo za standardního nastavení a běhu systému s jediným uživatelem spuštěným programem, kterým byl MATLAB.

První rychlostní test byl proveden na jednom z krátkých dostupných videí. Testované video mělo 23 vteřin (552 snímků). Všechny metody byly aplikovány pro každý jeden snímek videa. Průměrné výsledky z několika běhů uvažovaných metod jsou vidět v tabulce 9.1.

Nejpomalejší přístupem se ukázalo být rozdělování a slučování regionů. Tato metoda by se dala zrychlit ořezáním obrazu pouze na určitou část v okolí bodu, kde bylo klasifikováno prase. Problémem by mohl být špatný prvotní úsudek. Ten by však šel vytvořit ze shody více segmentačních metod. Nejrychlejší metodou byla segmentace prahováním s využitím prahu vypočteného Otsuovou metodou. Přibližně stejně rychlá byla metoda odečítání pozadí. Z experimentů se ukázalo, že metoda odečítání pozadí je ale spolehlivější a úspěšně odfiltruje světlé regiony obrazu, které jsou způsobeny odleskem nejspíše na čočkách kamer. Toho je dosaženo z důvodu stálosti odlesků. Odlesky tak byly obsaženy i v uvažovaném modelu pozadí a z aktuálního snímku byly odečteny.

Kvůli své rychlosti a velké přesnosti segmentace byla pro bílá prasata jako hlavní metoda pro získání regionu prasete zvolena právě metoda využívající odečítání pozadí. Takto získaný region byl poté podroben analýze prokládání elipsou nebo nastavením bodového modelu. V tabulce 9.1 je vidět, že zvolená metoda prokládání elipsy je dostatečně rychlá i pro zpracování obrazu v reálném čase.

Metoda	průměrná doba zpracování
Otsuova metoda	5.3 s
K-Means shlukování	145 s
Rozdělování a slučování regionů	1900 s
Odečítání pozadí	6.5 s
Odečítání pozadí s prokládání elipsou	7.7 s
Odečítání pozadí s PDM	60 s
Mediánový model pozadí	26.5 s
Modelování pomocí směsi gausiánů	350 s
Optický tok podle Horna a Schuncka	1100 s
Optický tok podle Liu	500 s
Kernel-based sledování	15 s

**Tabulka 9.1.** Rychlost testovaných metod pro video o 552 snímcích.

Metoda analýzy s pomocí bodového modelu již zpracování zpomalí znatelně.

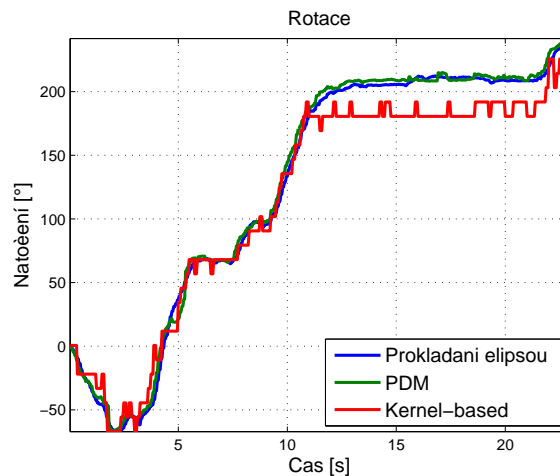
Výsledky prvního testu jsou vidět na obrázku č.9.1. Je vidět, že všechny metody dokázaly prase více či méně dobře sledovat.

Největší problém byl pozorován u sledování pohybu s použitím metod zaznamenávajících rozdíl mezi snímky (např. optický tok nebo mediánové modelování pozadí). To je způsobeno faktem, že při zpracování 24 snímků za vteřinu není mezisnímková změna tak velká a proto dochází ke špatným detekcím. Např. pokud prase hýbe pouze hlavou, zaznamená se takový pohyb a jeho pozice, když poté začne hýbat i zadní částí těla, zaznamená se pozice zadní části těla. Prase je stále na stejném místě, pohyb může ale být v blízkých snímcích zaznamenán i s velkým rozdílem.

Proto je ke konci videa pozorovatelný rozdíl v ustálené poloze prasete. Metody, které segmentovali prase velmi přesně (např. odečítání pozadí) při minimálním pohybu prasete dávají jako výstupní pozici souřadnice jeho středu. Metody zakládající se na změně mezi snímky mohou uvíznout na hlavě nebo zadní části prasete.

Pro zvětšení rychlosti se zachováním dostatečné přesnosti (u některých metod se přesnost dokonce zvýší) bylo využito zvýšení rozestupu mezi zpracovávanými snímky. Nezpracovávalo se tedy všech 24 každou vteřinu ale méně. Počet přeskočených snímků v každé vteřině videa je pro každou metodu jiný a byl určen experimentálně. Časová náročnost se v závislosti na počtu vynechaných snímků přibližně lineárně snižuje.

Toto omezení zvyšuje přesnost při zpracování videa metodami založenými na optických tocích, protože uvažované metody využívají rozdíl mezi snímky, který je při delším časovém rozestupu mezi zpracovávanými snímky větší.



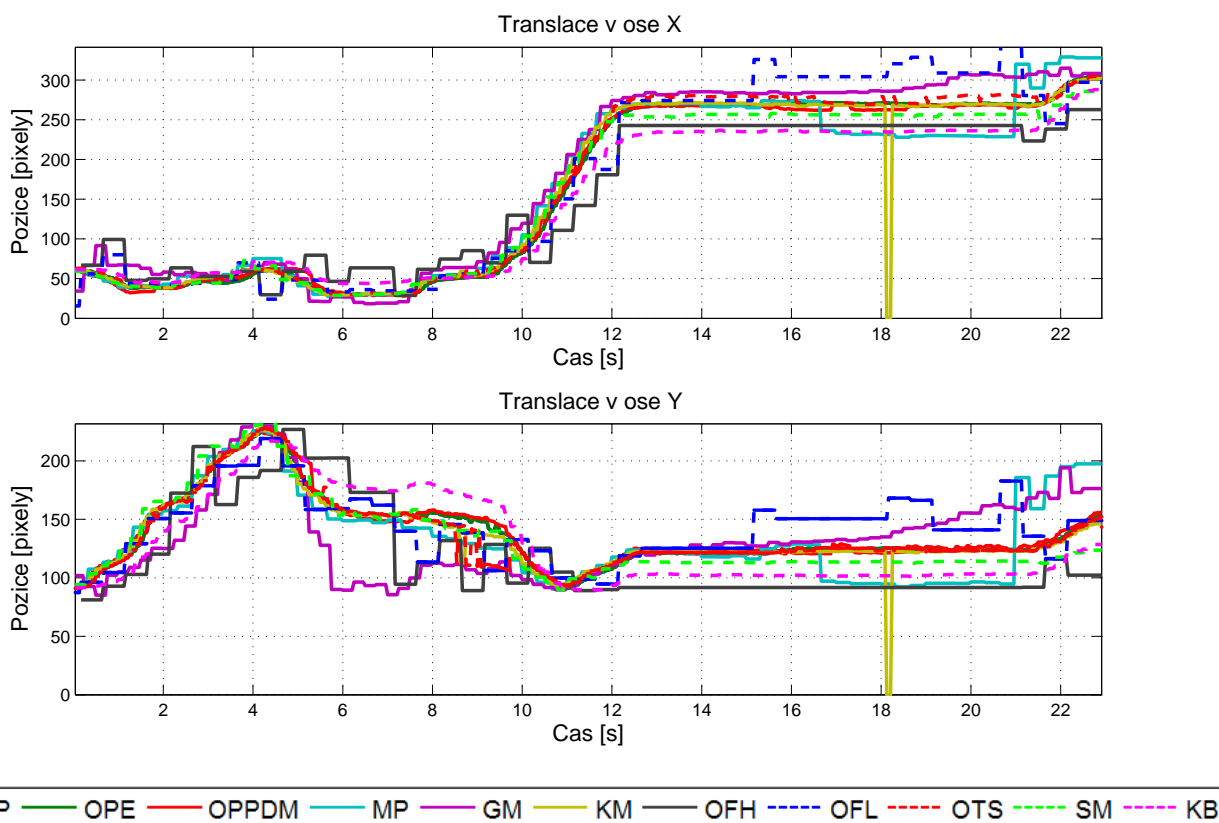
**Obrázek 9.2.** Srovnání naměřených rotací prasete pro krátké video.

Úhel natočení prasete byl rozpoznán velmi přesně všemi metodami. To je doloženo na obr. č. 9.2.

Kontrola práce všech metod byla prováděna zkontrolováním obrůzků, které jsou generovány do výstupní složky po určitém kroku snímků (pouze jednotky či desítky za jedno video).

Na výsledcích z takto krátkého videa je vidět, že nejlepším přístupem k segmentaci bílých prasat je v tomto případě odečítání pozadí. Velmi dobře fungovala také metoda využívající kernel-based sledování, oproti odečítání pozadí však není tak rychlá. U obou je navíc snadno získatelná rotace prasete.

Známé a často používané metody sledování pohybu na základě průběžně obnovovaného pozadí se ukázaly jako poměrně nepřesné a neúspěšné. To je dáno stálou přítomností prasete v obraze. Vhodnější úlohou pro použití těchto metod by bylo například



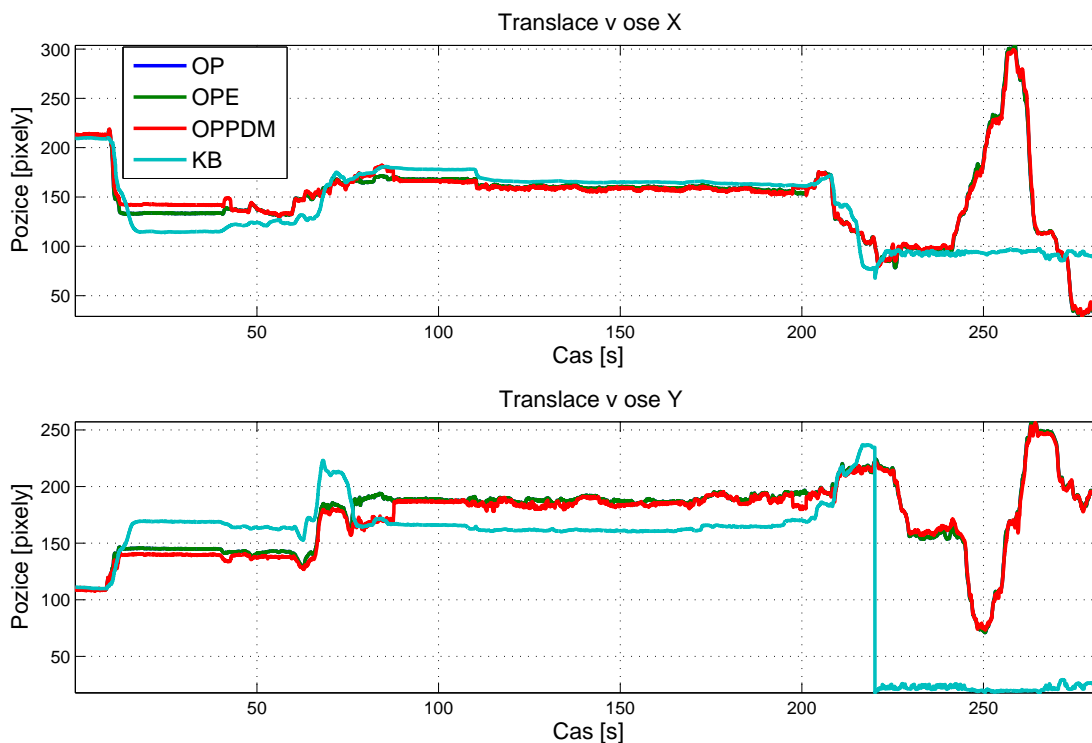
**Obrázek 9.1.** Srovnání naměřených pozic prasete v jednotlivých osách pro krátké video. Doplnění legendy: OP - odečítání pozadí, OPE - odečítání pozadí s proložením dat elipsou, OPPDM - odečítání pozadí s proložením bodovým modelem, MP - mediánový model pozadí, GM - směs gaussiánů, KM - k-Means shlukování, OFH - optický tok podle Horna a Schuncka, OFL - optický tok podle Liu, SM - rozdělování a slučování regionů, KB - Kernel-based sledování

sledování automobilů na dálnici ze statické kamery. Tedy takové situace, kde se hledaný objekt vyskytuje před kamerou pouze krátce.

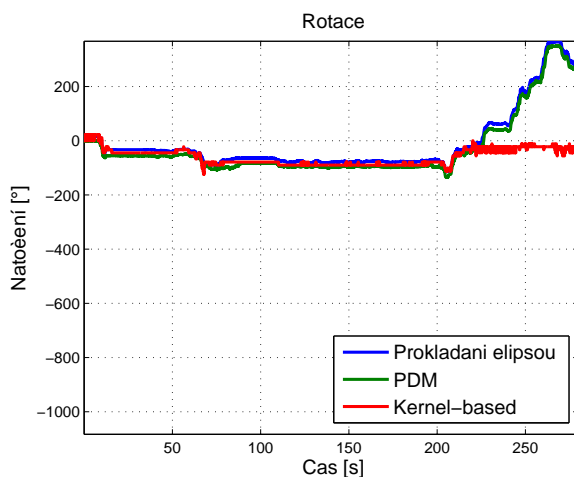
Od sledování zvířete s využitím optických toků byly očekávány dobré výsledky. Metoda, kterou popisuje Liu [18] se v pokusech s malým počtem obrázků ukázala jako docela přesná. Problémem ovšem je zvolení vhodné prahu velikostí výsledných regionů tak, aby pohyby hlavy a zadní části prasete byly odfiltrovány a v potaz se bral opravdu pouze jen pohyb celého prasete. Zároveň ale není žádoucí, aby byly odfiltrovány pohyby celého prasete, které se pohne mezi dvěma uvažovanými snímky pouze o kousek.

Nejllepšími metodami byla zpracována i pětiminutová videa. Pomocí odečítání pozadí je pět minut videa zpracováno za 75 vteřin, využití prokládání elipsou prodlouží zpracování o dalších 6 vteřin, bodový model už natáhne analýzu na 600 vteřin. Kernel-based sledování trvá pro pětiminutové video 85 vteřin. To je méně než bylo odhadováno při implementaci této metody. Ukázkové výsledky pro zpracování pětiminutového videa bílého prasete jsou vidět na obrázcích č. 9.3 a č. 9.4.

Jak výsledky ukazují, Kernel-based sledování má ke konci videa problémy. To je způsobeno rozdílnou hladinou intenzity osvětlení v kotci. Prase není ve všech částech kotce osvětleno stejně. Bylo by třeba aktualizovat model pro sledování prasete, problémem však je volba vhodného okamžiku k aktualizaci modelu. Pokud by ve snímku zvoleném pro aktualizaci nebylo prase detekováno zcela přesně, takto zanesenou chybu by už bylo těžké opravit a propagovala by se do dalšího sledování.



**Obrázek 9.3.** Srovnání naměřených pozic prasete v jednotlivých osách. Doplnění legendy: OP - odečítání pozadí, OPE - odečítání pozadí s proložením elipsou, OPPDM - odečítání pozadí s proložením bodovým modelem, KB - Kernel-based sledování



**Obrázek 9.4.** Srovnání naměřených rotací prasete.

Protože je při zpracování velkého množství videí požadována co největší přesnost a rychlost zpracování, byla pro hromadné zpracování všech videí bílých prasat nakonec použita metoda odečítání pozadí.

Tento přístup využívající předem známý model pozadí ale nelze použít u hnědého prasete. Prase moc splývá s okolím a kvalita videa byla u tohoto prasete nejhorší ze všech. To bylo nejspíše ovlivněno špatnou pozicí kamery a špatným stíněním. Příklad špatně zaznamenaného snímku je vidět na obr. č. 9.5. Ze stejného důvodu nelze použít ani sledování využívající k-Means shlukování.



**Obrázek 9.5.** Ukázka špatně detekovatelného hnědého prasete Bruna.

Nejlepším se ukázalo být takové sledování pohybu, které využívá změny mezi snímky. V případě hnědého prasete bylo možné získat poměrně přesné výsledky metodou využívající optické toky. Z dvou zmíněných metod v kapitole č. 6 je přesnější a rychlejší metoda podle Liu [18]. Ani touto metodou však není prase segmentováno tak dobře, aby mohl být pro analýzu pohybu použit bodový model. Prokládání segmentovaných dat elipsou také není ve 100% případech přesné, protože se může stát, že region odpovídající praseti bude segmentován ve špatném tvaru a elipsa bude umístěna mimo prase se špatným úhlem natočení.

Kernel-based sledování také využívá rozdílů mezi snímky, je však problém jak s inicializací metody tak s jejím průběhem. Na vině je opět přílišná podobnost prasete a okolí. Protože posun prostorové masky je založen na histogramu, je sledování objektu s podobnou intenzitou jako pozadí problematické.

Z metod s postupným modelováním pozadí prezentovaných v kapitole 5.2 fungovala pro hnědé prase lépe mediánová metoda. Její model se ukázal být věrohodnějším.

Z důvodu přílišné časové náročnosti a velké citlivosti na zadání konstanty byly zavrhnuty metody přímé konstrukce regionů (rozdělování a slučování).



# Kapitola 10

## Hromadné zpracování videa

### 10.1 Implementace

Pro pohodlné hromadné zpracování videa v prostředí MATLAB, bylo navrženo a vytvořeno jednoduché grafické uživatelské rozhraní právě v prostředí MATLAB. Toto rozhraní je vidět na obr. č. 10.1.

Vytvořené rozhraní dává uživateli možnost zvolit si sadu videí a to jak po jednom až po několika konkrétních videích, tak i celé složky. Dále je k dispozici možnost výběru specifického času, ve kterém se mají videa zpracovávat. Tato volba je dostupná kvůli tomu, aby bylo rychlejší a jednodušší spustit hromadnou analýzu videí pouze v částech, které jsou v danou chvíli důležité. Při analýze nahraných videí se klade důraz na chování prasat v době krmení. Zvolením konkrétních časů se dají nepotřebná videa lehce odfiltrvat.

Zvolená videa jsou zpracována všemi metodami, které skript spouštějící analýzu nalezne v podsložce *Metody*. Všechny metody musí začínat jménem prasete, pro které je tato metoda určena, odděleným podtržítkem od zbytku řetězce názvu metody.

Výstup použitých metod sledování je uložen do složky s názvem specifikovaným v příslušném textovém poli. Výstupní strom složek je formátován následujícím způsobem: *Jmeno složky s vystupem/jmeno prasete/jmeno metody/datum nahrání videa/jmeno videa/výstupní soubory*. Výstupem z každého videa je několik .png obrázků pro kontrolu práce algoritmu při sledování zvířete, .mat soubory s průběhem pozic na ose x a y, případně .mat soubor s průběhem rotace prasete a .mat soubory s časovými značkami ve formě počtu vteřin od začátku videa a unixové časové značce. Unixová časová značka je počet vteřin od 1.1.1970 0:00:00 koordinovaného světového času.

Protože se ve videích vyskytuje více snímků odpovídajících jedné vteřině, je zapisovaná unixová značka rozšířena o desetinnou část, které odpovídá příslušnému intervalu mezi poslední celou vteřinou a aktuálním časem snímku.

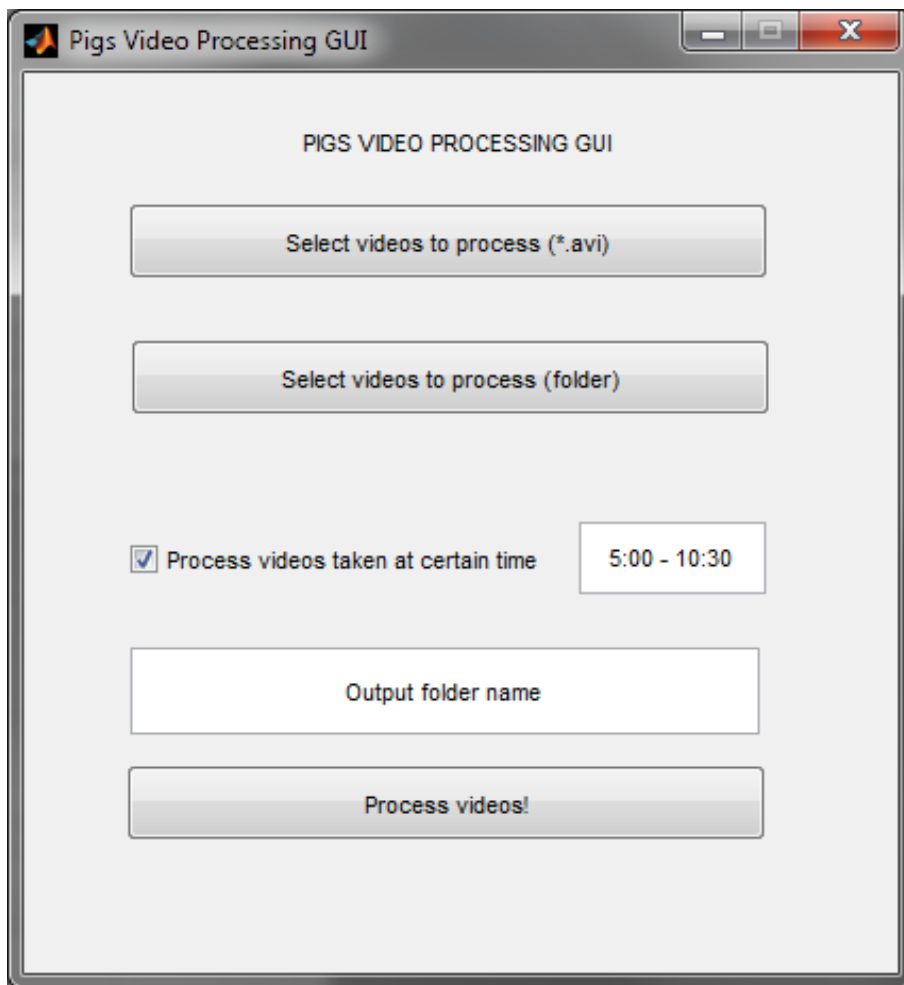
Při zpracování zadaných vstupů bylo využito dvou převzatých metod – metoda pro rekurzivní procházení složek *dirrec*<sup>1)</sup> a metoda pro exportování obrázků *exportfig* z [16].

Při zpracování videí metodou odečítání pozadí bylo možné zpracovat celou sadu videí pro prase Alf (téměř 2400 videí a průměrně přibližně 7000 snímků v každém videu) i s prokládáním elipsou, generováním kontrolních obrázků a průběhů za 48 hodin. Při zpracování ale záleží i na množství generovaných kontrolních dat.

### 10.2 Klasifikace pohybu

Výsledky zpracování pohybu, jednotlivé průběhy translace a rotace prasete v čase, byly klasifikovány na části, kde se prase hýbe a části bez pohybu prasete. Rozdělení do těchto dvou skupin bylo provedeno na základě velikosti rozptylu uvažovaného parametru v časovém okně předem známé délky. Za zkoumaný parametr byla zvolena jak rychlost

<sup>1)</sup> <http://www.mathworks.fr/matlabcentral/fileexchange/15505-recursive-dir>



**Obrázek 10.1.** Grafické uživatelské rozhraní v prostředí MATLAB pro pohodlné zpracování videí.

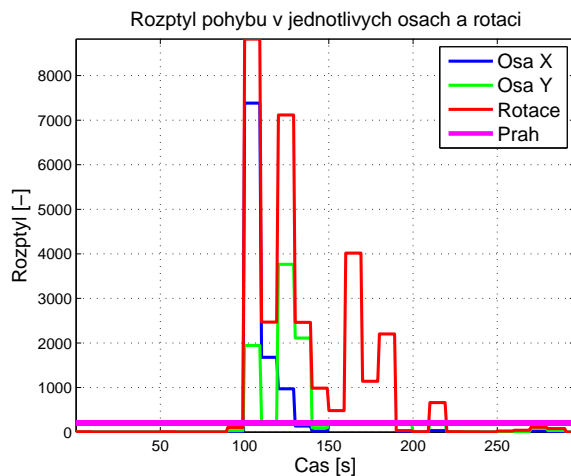
pohybu (derivace polohy) tak také samotná poloha. Z experimentů se ukázalo, že přesnější klasifikace je dosaženo při volbě průběhu polohy než při volbě rychlosti. Křivka průběhu rychlosti není tak hladká jako křivka průběhu polohy.

Křivky poloh v jednotlivých osách jsou zpracovány po oknech předem známé délky. Vzhledem k tomu, že prase je nahráváno po dlouhou dobu a záměrem není zjistit přesný počet vteřin pohybu či klidu, bylo zvoleno okno délky 10 vteřin. V tomto okně byl pro všechny uvažované průběhy vypočítán rozptyl signálu a po jeho porovnání s předem zadaným prahem bylo určeno, zda se prase v tomto období hýbalo či bylo neaktivní.

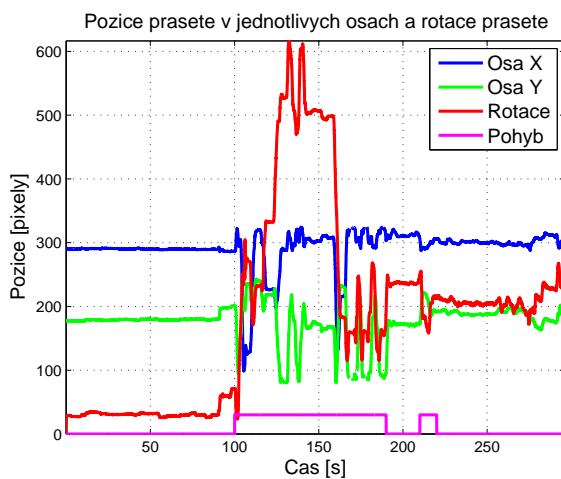
Výsledky klasifikace pro náhodně zvolené video jsou vidět na obr. č. 10.2 a č. 10.3.

Klasifikace na pohyb a klid se u všech testovaných videí ukázala jako úspěšná. Záleží pouze na tom, jak jsou nastaveny prahy pro rozlišení mezi oběma stavy. Pokud je zájem pouze o pohyb prasete po celém kotci, lze práh zvýšit, pokud je třeba rozlišit i popocházení na místě (např. když prase jí), práh může být snížen. Poměr pohybu a klidu v každém videu byl pro možnost další analýzy zapsán do textového souboru.

Ukázka výstupních klasifikačních hodnot pro delší časové úseky je prezentována v příloze B.



Obrázek 10.2. Rozptyly průběhů jednotlivých pohybů a použitý práh.



Obrázek 10.3. Klasifikace pohybu (pro hodnotu  $pohyb > 0$  byl klasifikován pohyb, pro  $pohyb = 0$  byl klasifikován klid).

# Kapitola 11

## Diskuze výsledků

Představenými metodami byly zpracovány stovky videí všech tří prasat.

Výsledky zpracování hnědého prasete jsou zcela jistě ovlivněny nevhodnou pozicí kamery nad kotci. Do záznamu se promítají neurčité odrazy a v dolní části videa je obraz velmi rozostřený. Vzhledem k tomu, že má hnědé prase podobnou barvu jako okolní podestýlka, je velmi ztížena správná detekce v této části obrazu. Odrazy jsou patrné i u videí bílých prasat. Pokud by byly kamery v budoucnu lépe umístěny, šlo by bílá prasata segmentovat zdaleka nejrychlejší metodou – prahováním.

Výhodou nahraných videí byla i skutečnost, že se velmi málo měnily kotce, ve kterých byla prasata umístěna a proto šlo pro bílá prasata jednoduše použít metodu odečítání pozadí. Tento fakt by se měl i do budoucna zachovat.

V kotcích by se ale před dalším nahráváním drobná změna provést mohla. Při sledování bílých prasat by bylo výhodné přetříť vnější strany zdí na jinou než bílou barvu. Opět by se pak zvýšila přesnost a jednoduchost prahování a celé zpracování a segmentace by mohly probíhat až o řád rychleji. Stejně tak by bylo vhodné více odlišit kotec a podestýlku v kotci, kde se vyskytuje hnědé prase.

Některé metody běží relativně pomalu také kvůli nesprávně optimalizovanému kódu pro prostředí MATLAB. Větší využívání funkcí pro práci s vektory, které jsou dodávány přímo v MATLABovské distribuci, by mohlo mít za výsledek razantní zlepšení co do rychlosti metod jako jsou rozdělování a slučování regionů či kernel-based sledování. Velkým přínosem pro rychlost zpracování videa by bylo odproštění se od prostředí MATLAB a přepsání kódu např. do jazyka C.

Obdržené výsledky jsou ale velmi dobré a alespoň u bílých prasat přesně sledují pozici i rotaci prasete a to s dodržением nízké časové náročnosti. U hnědého prasete je s dostatečnou přesností a spolehlivostí možné sledovat pouze translaci.

Největším nedostatkem předvedených metod je špatné rozlišení částí prasete, které jsou barevně velmi podobné podestýlce kotců. Příkladem může být hlava prasete. Detekce co největší části prasete by pomohla k lepšímu sledování jeho pohybu.

Navržené metody si také neporadí s drobnými pohyby, které by se mohli u prasat s Huntingtonovou nemocí vyskytnout. Nelze například dostatečně přesně rozlišit chvění prasete, pohyb ocasu ani přesně určit pohyb hlavy.

Metoda klasifikace naměřených průběhů pohybu prasete fungovala přesně a výsledky měly vypovídající hodnotu o množství pohybu prasete po kotci v jednotlivých částech dne.

# Kapitola 12

## Závěr

Cílem této práce bylo zajistit hromadné zpracování dodaných videí a vhodným způsobem z nich získat pohyb prasete. Videá byla hromadně úspěšně zkonvertována do standardního formátu a bylo vytvořeno grafické uživatelské rozhraní do prostředí MATLAB, pomocí kterého bylo zajištěno zpracování a analýza právě v prostředí MATLAB.

Byly popsány, implementovány a vzájemně srovnány známé metody segmentace obrazu a sledování pohybu objektu ve videu. Nejlepší z nich byly použity pro dávkové zpracování dodaných videí.

Nejpřesnějších výsledků a velké rychlosti dosahovala pro bílá prasata metoda využívající odečítání předem známého modelu pozadí. Výstup této metody byl dále použit pro analýzu proložením dat elipsou. Implementované řešení bylo velmi rychlé a dodalo další parametr pohybu zvířete – rotaci. Také bylo implementováno a otestováno sledování pohybu pomocí bodového (tzv. point-distribution) modelu. Tato metoda rovněž využívá výstup segmentace obrazu pomocí odečítání pozadí. Podrobnější analýza bodového modelu nakonec nebyla provedena, metoda je však připravena k budoucímu použití.

Z experimentů se ukázalo, že nejpoužitelnější metodou pro sledování hnědého prasete je sledování s využitím optických toků. Zpracování videí hnědého prasete však mělo vyšší časovou náročnost.

Byly navrženy opatření, kterými by se dalo zlepšit a zrychlit zpracovávání videí. Příkladem může být přetření zdí kotců nebo lehké upravení polohy či stínění kamer.

Na výstupních datech ze zpracování obrazu byla provedena klasifikace pohybu prasete v kotci na část, kdy se prase hýbe, a část bez pohybu. To je důležitý ukazatel dlouhodobého vývoje aktivity prasete. Tato klasifikace proběhla s dobrými výsledky.

Získané výsledky budou použity v dalším výzkumu s behaviorálními zvířecími modely Huntingtonovy choroby.

## Literatura

- [1] I. L. Andersen, G. Færevik, K. E. Bøe, A. M. Janczak, and M. Bakken. Effects of diazepam on the behaviour of weaned pigs in three putative models of anxiety. *Applied animal behaviour science*, 68(2):121–130, 2000.
- [2] M. Mikkelsen, A. Møller, L. H. Jensen, A. Pedersen, J. B. Harajehi, and H. Pakkenberg. Mptp-induced parkinsonism in minipigs: a behavioral, biochemical, and histological study. *Neurotoxicology and teratology*, 21(2):169–175, 1999.
- [3] D. C. Rubinsztein. Lessons from animal models of huntington’s disease. *TRENDS in Genetics*, 18(4):202–209, 2002.
- [4] M. Baxa, M. Hruska-Plochan, S. Juhas, P. Vodicka, A. Pavlok, J. Juhasova, A. Miyanochara, T. Nejime, J. Klima, M. Macakova, et al. A transgenic mini-pig model of huntington’s disease. *Journal of Huntington’s Disease*, 2(1):47–68, 2013.
- [5] J. Kongstro. Development of a computer vision system to monitor pig locomotion. *Open Journal of Animal Sciences*, 3(3):321–325, 2013.
- [6] M. A. Kashiha, C. Bahr, S. Ott, Ch. P. H. Moons, T. A. Niewold, F. Tuytens, and D. Berckmans. Automatic monitoring of pig activity using image analysis. In J. Blanc-Talon, A. Kasinski, W. Philips, D. Popescu, and P. Scheunders, editors, *Advanced Concepts for Intelligent Vision Systems*, volume 8192 of *Lecture Notes in Computer Science*, pages 555–563. Springer International Publishing, 2013.
- [7] N. M. Lind, M. Vinther, R. P. Hemmingsen, and A. K. Hansen. Validation of a digital video tracking system for recording pig locomotor behaviour. *Journal of neuroscience methods*, 143(2):123–132, 2005.
- [8] J. A. Marchant and Ch. M. Onyango. Fitting grey level point distribution models to animals in scenes. *Image and vision computing*, 13(1):3–12, 1995.
- [9] A. M. Baumberg and D. C. Hogg. An efficient method for contour tracking using active shape models. In *Motion of Non-Rigid and Articulated Objects, 1994., Proceedings of the 1994 IEEE Workshop on*, pages 194–199. IEEE, 1994.
- [10] T. F. Cootes, Ch. J. Taylor, D. H. Cooper, and J. Graham. Training models of shape from sets of examples. In *BMVC92*, pages 9–18. Springer, 1992.
- [11] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 142–149. IEEE, 2000.
- [12] H. Zhang, Z. Huang, W. Huang, and Liyuan Li. Kernel-based method for tracking objects with rotation and translation. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, pages 728–731. IEEE, 2004.
- [13] M. Sonka, V. Hlavac, R. Boyle, et al. *Image processing, analysis, and machine vision*, volume 3. Thomson Toronto, 2008.

- 
- [14] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.
  - [15] Ch. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.
  - [16] T. Svoboda, J. Kybic, and V. Hlavac. *Image Processing, Analysis & Machine Vision-A MATLAB Companion*. Thomson Learning, 2007.
  - [17] B. K. Horn and B. G. Schunck. Determining optical flow. In *1981 Technical Symposium East*, pages 319–331. International Society for Optics and Photonics, 1981.
  - [18] C. Liu et al. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, Massachusetts Institute of Technology, 2009.
  - [19] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564–577, 2003.
  - [20] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):476–480, 1999.







# Příloha **A**

## Obsah přiloženého CD

- **/src** - zdrojové kódy do prostředí MATLAB
- **/text** - PDF verze této BP
- **/video** - ukázková videa

## Příloha B

### Ukázka klasifikovaného pohybu pro jednotlivé dny

Č.	den	pohyb [%]	klid [%]
1	01. 02. 2011	62.6334	37.3666
2	02. 02. 2011	32.2815	67.7185
3	03. 02. 2011	35.0579	64.9421
4	04. 02. 2011	22.9969	77.0031
5	05. 02. 2011	21.5156	78.4844
6	15. 02. 2011	30.6394	69.3606
7	16. 02. 2011	49.6886	50.3114
8	17. 02. 2011	25.7403	74.2597
9	18. 02. 2011	35.8532	64.1468
10	19. 02. 2011	22.7845	77.2155
11	20. 02. 2011	28.4761	71.5239
12	21. 02. 2011	27.4282	72.5718
13	22. 02. 2011	31.064	68.936
14	23. 02. 2011	13.3231	86.6769
15	24. 02. 2011	29.7342	70.2658
16	25. 02. 2011	31.0665	68.9335
17	26. 02. 2011	25.5646	74.4354
18	27. 02. 2011	72.7144	27.2856
19	28. 02. 2011	32.8104	67.1896
20	01. 03. 2011	45.937	54.063
21	02. 03. 2011	15.5837	84.4163

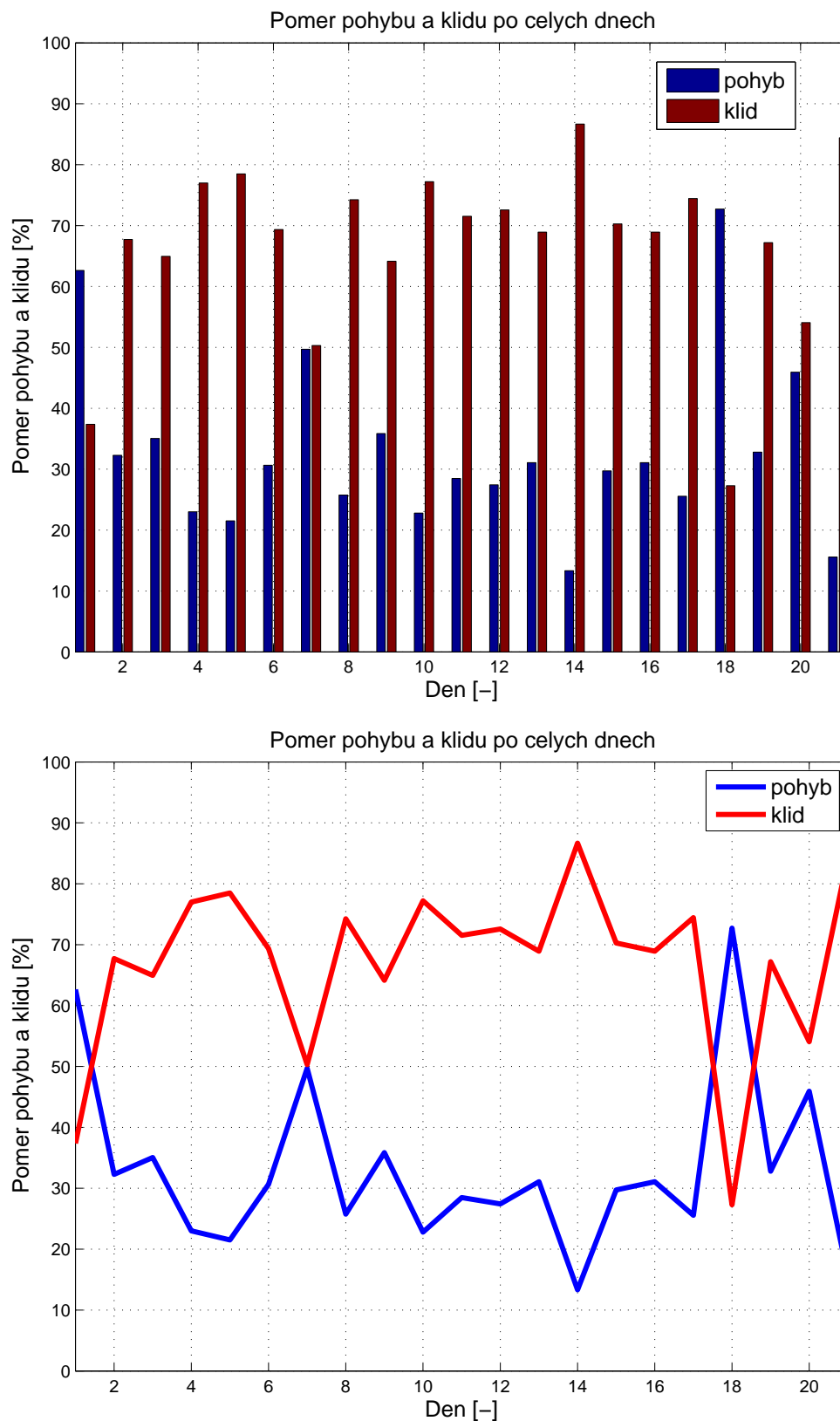
**Tabulka B.1.** Klasifikovaný poměr pohybu a klidu videa prasete Alf v jednotlivých dnech

Pozn.: Hodnota *NaN* (not a number) znamená, že v daný den nebyla videa v určitou denní dobu nahrávána.

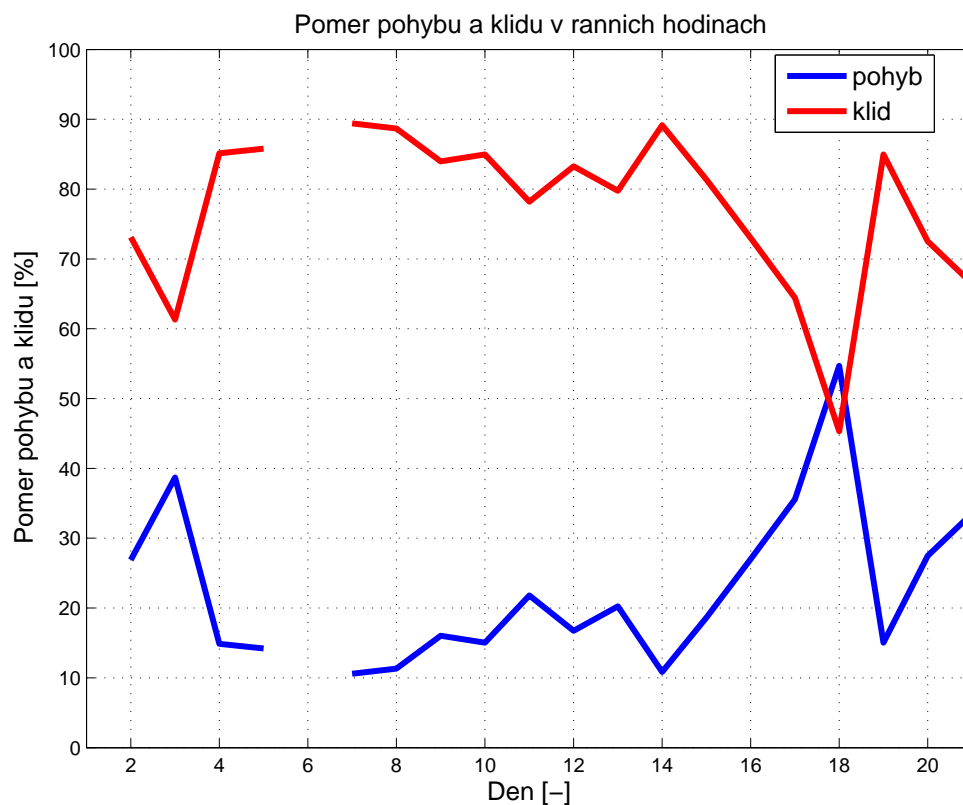
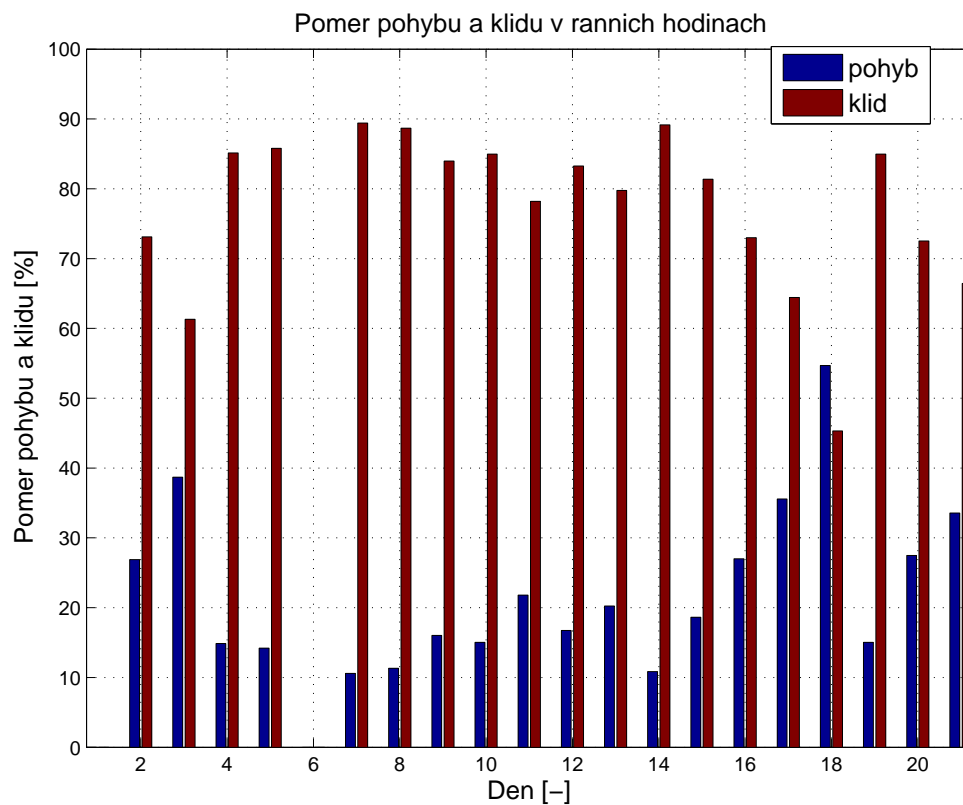
Pozn.: Zobrazené hodnoty pro prase Babe (druhá série) jsou z videí nahraných pouze do 15:30.

Č.	den	pohyb r. [%]	klid r. [%]	pohyb odp. [%]	klid odp. [%]
1	01. 02. 2011	NaN	NaN	62.6334	37.3666
2	02. 02. 2011	26.884	73.116	33.361	66.639
3	03. 02. 2011	38.6967	61.3033	34.471	65.529
4	04. 02. 2011	14.8661	85.1339	24.0461	75.9539
5	05. 02. 2011	14.2018	85.7982	23.7098	76.2902
6	15. 02. 2011	NaN	NaN	30.6394	69.3606
7	16. 02. 2011	10.5802	89.4198	54.9031	45.0969
8	17. 02. 2011	11.3318	88.6683	27.5414	72.4586
9	18. 02. 2011	16.0218	83.9782	40.4805	59.5195
10	19. 02. 2011	15.0352	84.9648	24.0761	75.9239
11	20. 02. 2011	21.7917	78.2083	29.5902	70.4098
12	21. 02. 2011	16.749	83.251	29.4952	70.5048
13	22. 02. 2011	20.2289	79.7711	34.4856	65.5144
14	23. 02. 2011	10.8463	89.1538	13.9986	86.0014
15	24. 02. 2011	18.6264	81.3736	31.3885	68.6115
16	25. 02. 2011	27.0114	72.9886	31.8775	68.1225
17	26. 02. 2011	35.5792	64.4208	22.5602	77.4398
18	27. 02. 2011	54.6826	45.3174	78.725	21.275
19	28. 02. 2011	15.0398	84.9602	36.3645	63.6355
20	01. 03. 2011	27.4728	72.5272	49.5107	50.4893
21	02. 03. 2011	33.5479	66.4521	5.1045	94.8954

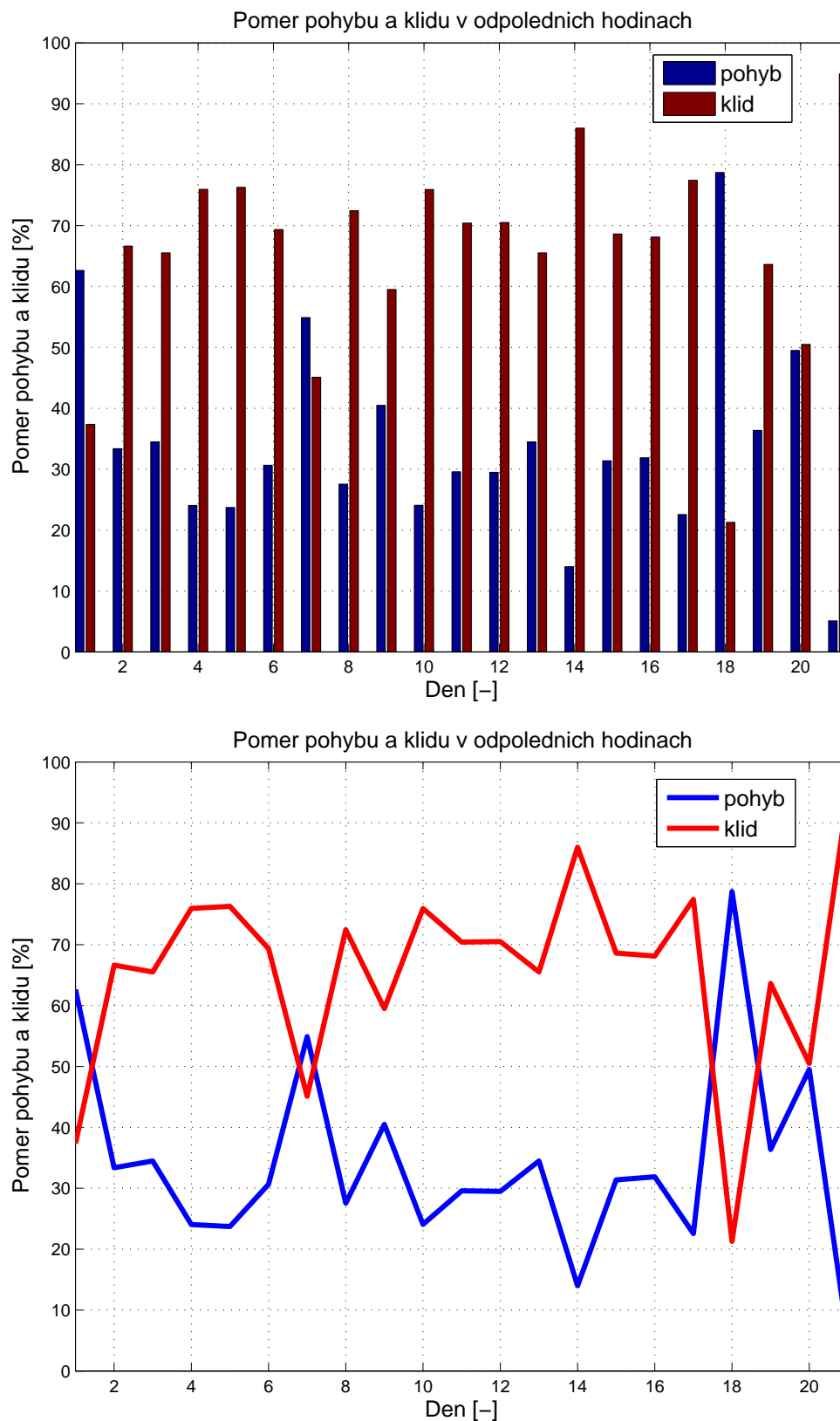
**Tabulka B.2.** Klasifikovaný poměr pohybu a klidu videa prasete Alf v jednotlivých dnech



Obrázek B.1. Poměr pohybu a klidu po celých dnech (číslo dne odpovídá dnům v předcházející tabulce).



Obrázek B.2. Poměr pohybu a klidu ráno (číslo dne odpovídá dnům v předcházející tabulce).



Obrázek B.3. Poměr pohybu a klidu odpoledne (číslo dne odpovídá dnům v předcházející tabulce).

Č.	den	pohyb [%]	klid [%]
1	02. 12. 2010	NaN	NaN
2	03. 12. 2010	41.378	58.622
3	04. 12. 2010	30.0308	69.9692
4	05. 12. 2010	56.0822	43.9178
5	06. 12. 2010	30.2664	69.7336
6	07. 12. 2010	55.5539	44.4461
7	08. 12. 2010	41.7126	58.2874
8	09. 12. 2010	31.7153	68.2847
9	10. 12. 2010	46.47	53.53
10	11. 12. 2010	36.9667	63.0333
11	12. 12. 2010	37.5366	62.4634
12	13. 12. 2010	50.3675	49.6325
13	14. 12. 2010	33.5936	66.4064
14	15. 12. 2010	42.935	57.065
15	16. 12. 2010	41.7723	58.2277
16	17. 12. 2010	27.6448	72.3552
17	18. 12. 2010	21.691	78.309
18	27. 12. 2010	38.6984	61.3016
19	28. 12. 2010	50.9897	49.0103
20	29. 12. 2010	52.2283	47.7717
21	30. 12. 2010	31.7486	68.2514
22	31. 12. 2010	34.6901	65.3099
23	01. 01. 2011	43.9956	56.0044
24	02. 01. 2011	40.9598	59.0402
25	03. 01. 2011	40.2147	59.7853
26	04. 01. 2011	33.3748	66.6252
27	05. 01. 2011	40.1108	59.8892
28	06. 01. 2011	28.9338	71.0662
29	07. 01. 2011	43.6982	56.3018
30	08. 01. 2011	34.4918	65.5082
31	09. 01. 2011	29.9812	70.0188
32	10. 01. 2011	39.6726	60.3274
33	11. 01. 2011	45.9048	54.0952
34	12. 01. 2011	47.9613	52.0387
35	14. 01. 2011	48.6393	51.3607
36	15. 01. 2011	40.1862	59.8138
37	26. 01. 2011	NaN	NaN
38	27. 01. 2011	55.3135	44.6865
39	28. 01. 2011	39.8576	60.1424
40	29. 01. 2011	47.3693	52.6307
41	30. 01. 2011	43.9582	56.0418
42	31. 01. 2011	13.134	86.866
43	01. 02. 2011	19.616	80.384
44	02. 02. 2011	9.5855	90.4145
45	03. 02. 2011	14.1514	85.8486

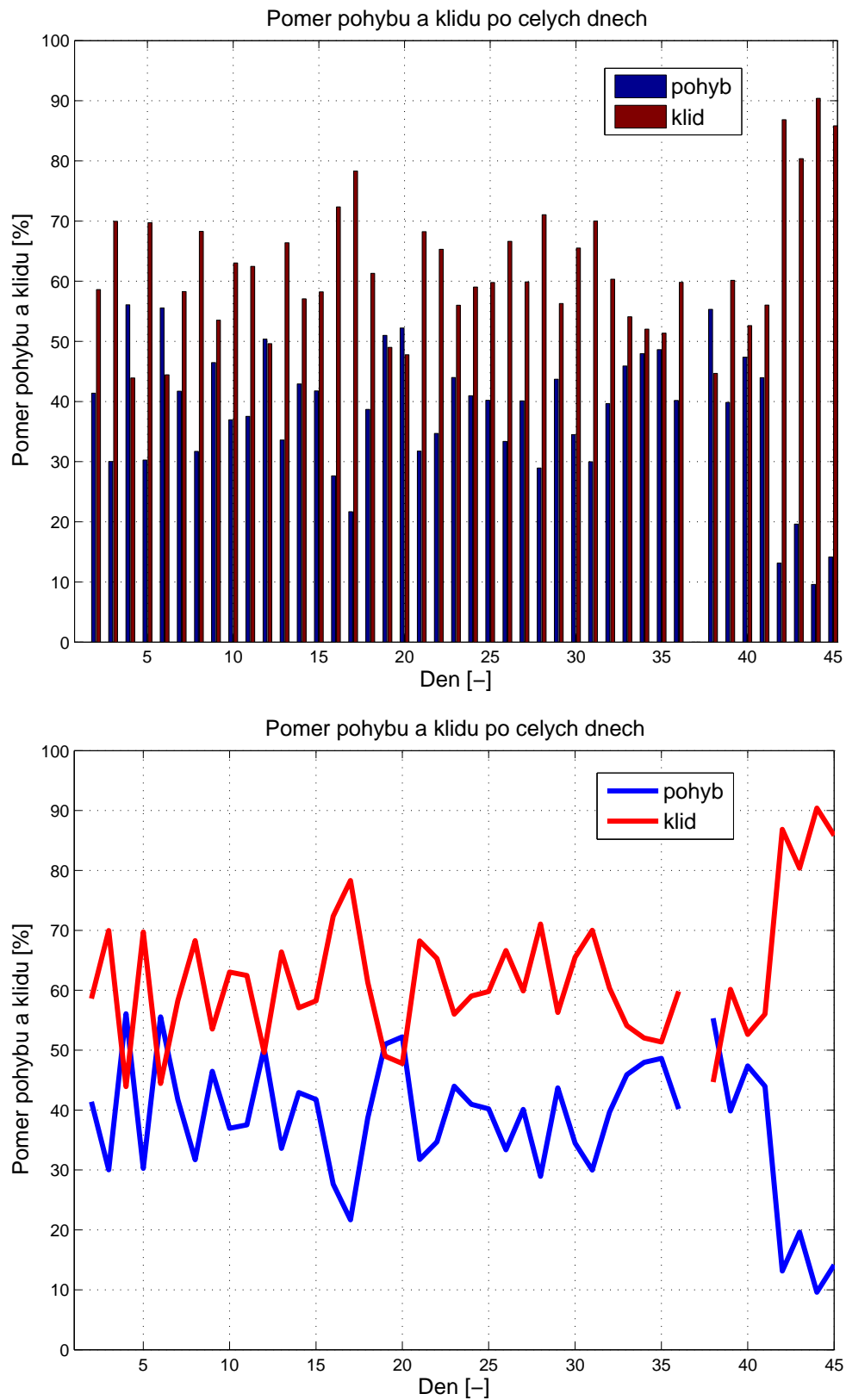
**Tabulka B.3.** Klasifikovaný poměr pohybu a klidu videa prasete Babe v jednotlivých dnech.



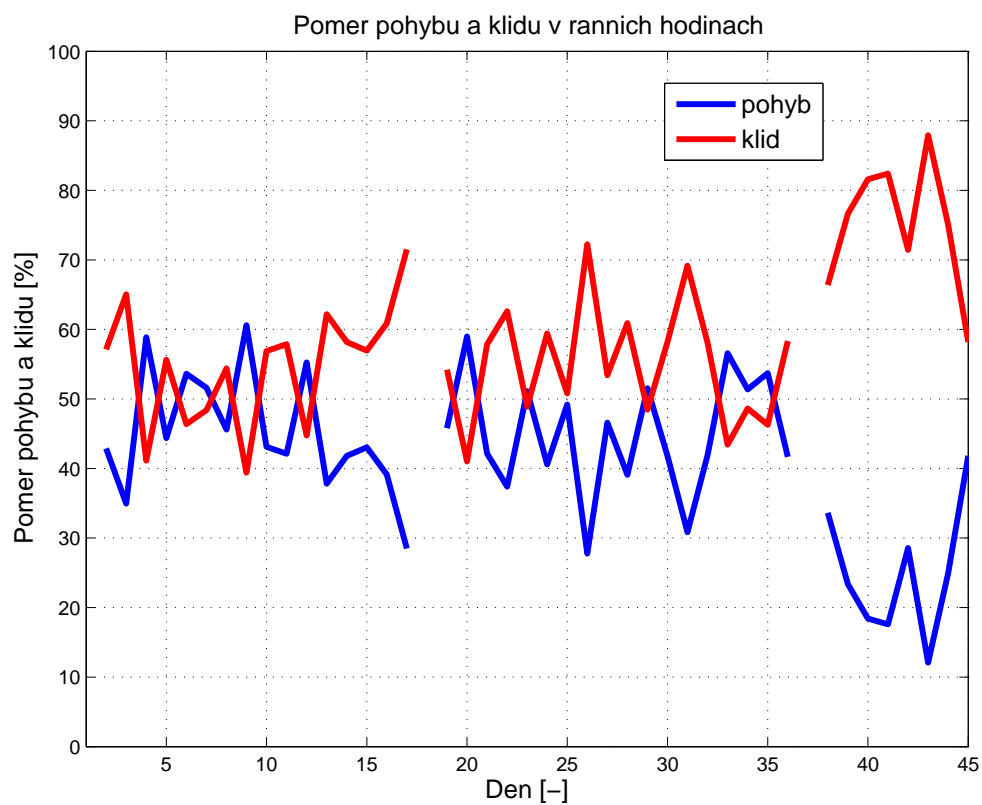
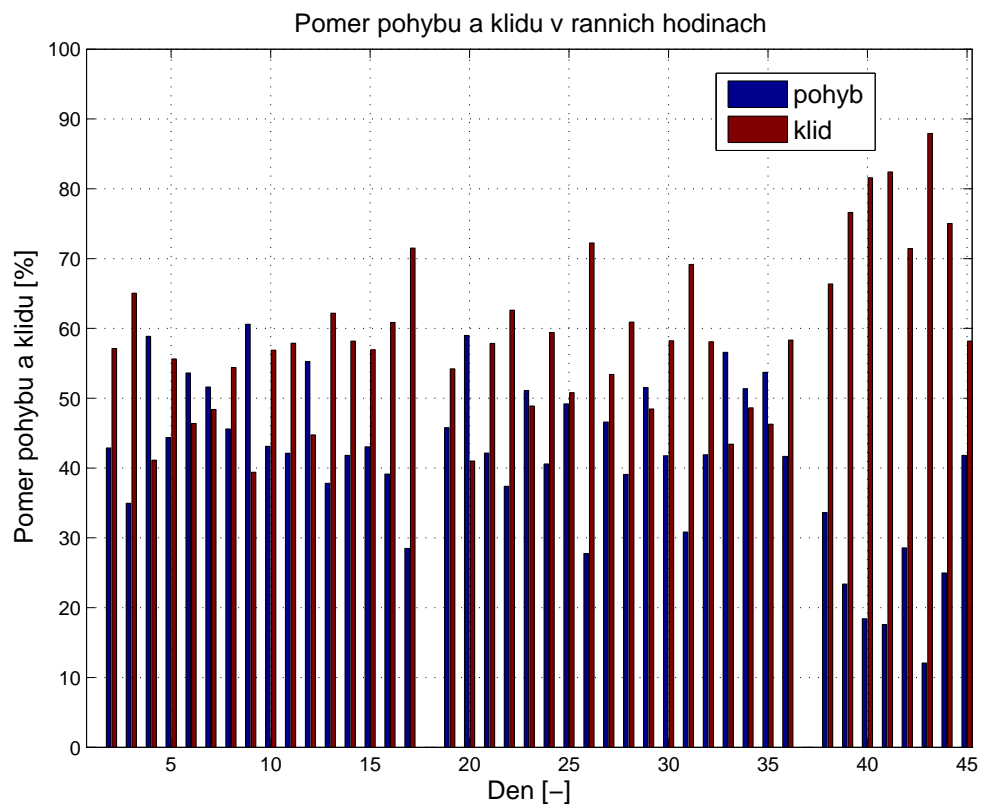


Č.	den	pohyb r. [%]	klid r. [%]	pohyb odp. [%]	klid odp. [%]
1	02. 12. 2010	NaN	NaN	NaN	NaN
2	03. 12. 2010	42.879	57.121	39.3766	60.6234
3	04. 12. 2010	34.9647	65.0353	23.0411	76.9589
4	05. 12. 2010	58.8797	41.1203	51.886	48.114
5	06. 12. 2010	44.3698	55.6302	10.2866	89.7134
6	07. 12. 2010	53.6209	46.3791	58.4533	41.5467
7	08. 12. 2010	51.6127	48.3873	27.6875	72.3125
8	09. 12. 2010	45.6069	54.3931	12.0354	87.9646
9	10. 12. 2010	60.609	39.391	26.4397	73.5603
10	11. 12. 2010	43.1159	56.8841	28.2554	71.7446
11	12. 12. 2010	42.1319	57.8681	31.0265	68.9735
12	13. 12. 2010	55.2508	44.7492	43.8564	56.1436
13	14. 12. 2010	37.8241	62.1759	27.2479	72.7521
14	15. 12. 2010	41.814	58.186	44.6166	55.3834
15	16. 12. 2010	43.0331	56.9669	39.9861	60.0139
16	17. 12. 2010	39.1311	60.8689	11.3725	88.6275
17	18. 12. 2010	28.4858	71.5142	14.8961	85.1039
18	27. 12. 2010	NaN	NaN	38.6984	61.3016
19	28. 12. 2010	45.7778	54.2222	57.9388	42.0612
20	29. 12. 2010	58.9904	41.0096	42.6487	57.3513
21	30. 12. 2010	42.156	57.844	17.0049	82.9951
22	31. 12. 2010	37.391	62.609	30.8638	69.1362
23	01. 01. 2011	51.1009	48.8991	33.3377	66.6624
24	02. 01. 2011	40.5833	59.4167	41.3677	58.6323
25	03. 01. 2011	49.1962	50.8038	27.4908	72.5092
26	04. 01. 2011	27.7736	72.2264	41.7766	58.2234
27	05. 01. 2011	46.5872	53.4128	30.3961	69.6039
28	06. 01. 2011	39.0919	60.9081	13.6966	86.3034
29	07. 01. 2011	51.5446	48.4554	32.5824	67.4176
30	08. 01. 2011	41.7659	58.2341	25.3993	74.6007
31	09. 01. 2011	30.8447	69.1553	28.8298	71.1701
32	10. 01. 2011	41.9152	58.0848	36.3089	63.6911
33	11. 01. 2011	56.5795	43.4205	29.8927	70.1073
34	12. 01. 2011	51.3632	48.6368	42.8585	57.1415
35	14. 01. 2011	53.7098	46.2902	41.0336	58.9664
36	15. 01. 2011	41.6629	58.3371	37.9712	62.0288
37	26. 01. 2011	NaN	NaN	NaN	NaN
38	27. 01. 2011	33.6288	66.3713	62.5418	37.4582
39	28. 01. 2011	23.38	76.62	43.5193	56.4807
40	29. 01. 2011	18.4088	81.5911	50.5872	49.4128
41	30. 01. 2011	17.5901	82.4099	49.8178	50.1822
42	31. 01. 2011	28.577	71.423	7.9864	92.0136
43	01. 02. 2011	12.0792	87.9208	22.547	77.453
44	02. 02. 2011	24.9609	75.0391	4.4604	95.5396
45	03. 02. 2011	41.8203	58.1797	4.9284	95.0716

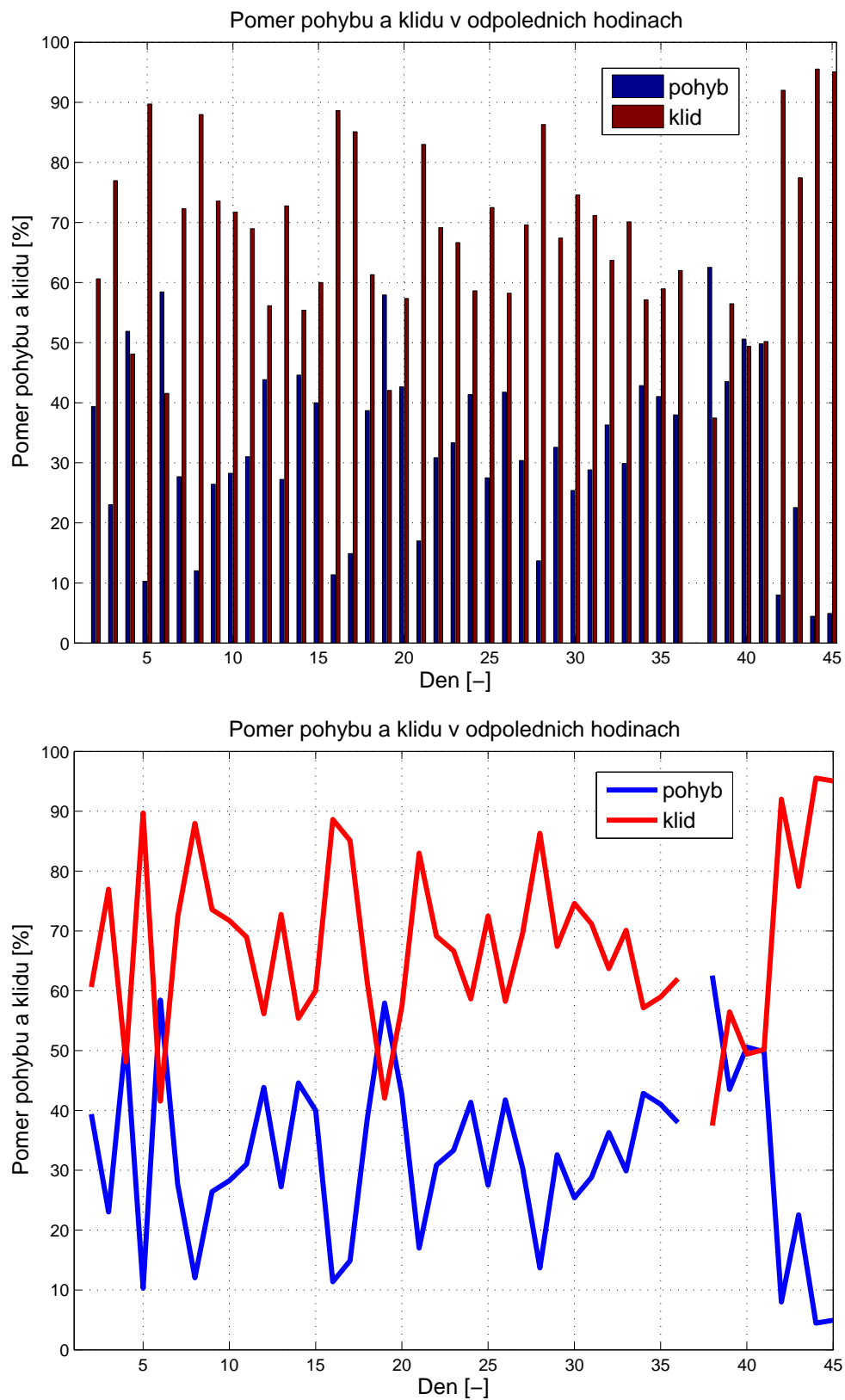
**Tabulka B.4.** Klasifikovaný poměr pohybu a klidu videa prasete Babe v jednotlivých dnech.



Obrázek B.4. Poměr pohybu a klidu po celých dnech (číslo dne odpovídá dnům v předcházející tabulce).



Obrázek B.5. Poměr pohybu a klidu ráno (číslo dne odpovídá dnům v předcházející tabulce).



**Obrázek B.6.** Poměr pohybu a klidu odpoledne (číslo dne odpovídá dnům v předcházející tabulce).