

**UNIVERZITA KOMENSKÉHO V BRATISLAVE**  
**FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY**

**DETEKCIA VÝZNAMNÝCH OBLASTÍ**  
**VO VIDEU**

**Diplomová práca**

**UNIVERZITA KOMENSKÉHO V BRATISLAVE**  
**FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY**

**DETEKCIA VÝZNAMNÝCH OBLASTÍ**  
**VO VIDEU**

**Diplomová práca**

Študijný program: Aplikovaná informatika  
Študijný odbor: 9.2.9. aplikovaná informatika  
Školiace pracovisko: Katedra Aplikovanej Informatiky  
Školiteľ: RNDr. Elena Šikudová, PhD.



Univerzita Komenského v Bratislave  
Fakulta matematiky, fyziky a informatiky

## ZADANIE ZÁVEREČNEJ PRÁCE

**Meno a priezvisko študenta:** Bc. Martin Kuchyňár  
**Študijný program:** aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)  
**Študijný odbor:** 9.2.9. aplikovaná informatika  
**Typ záverečnej práce:** diplomová  
**Jazyk záverečnej práce:** slovenský  
**Sekundárny jazyk:** anglický

**Názov:** Detekcia významných oblastí vo videu  
*Spatio-temporal salient object detection*

**Cieľ:** Metódy na detekciu významných oblastí vo videu:  
1. Naštudovanie  
2. Návrh zlepšenia  
3. Implementácia  
4. Porovnanie výsledkov

**Vedúci:** RNDr. Elena Šikudová, PhD.  
**Katedra:** FMFI.KAI - Katedra aplikovanej informatiky  
**Vedúci katedry:** doc. PhDr. Ján Rybár, PhD.  
**Dátum zadania:** 20.10.2014

**Dátum schválenia:** 24.10.2014

prof. RNDr. Roman Ďurikovič, PhD.  
garant študijného programu

  
.....  
študent

  
.....  
vedúci práce

## **Čestné vyhlásenie**

Čestne prehlasujem, že som túto diplomovú prácu vypracoval samostatne s použitím uvedených zdrojov.

V Bratislave

.....

## **Pod'akovanie**

Ďakujem mojej vedúcej práce RNDr. Elene Šikudovej, PhD. za odborné vedenie, profesionálne usmernenie a cenné rady, ktoré mi spokytla pri vypracovávaní diplomovej práce.

## **Abstrakt**

V diplomovej práci sú navrhnuté a zvalidované metódy pre spracovanie videa a detekciu významných oblastí vo videu. Pričom sa zameriava na získanie čisto dynamických príznakov, ktoré je možné následne kombinovať s klasickými príznakmi. Pre uľahčenie prototypovania podobných modelov, bola vytvorená ucelená aplikácia v prostredí matlab, poskytujúcu automatickú validáciu modelu pomocou štandardných datasetov a jednoduché vkladanie konkurenčných modelov pre okamžité porovnávanie. Práca je rozdelená do piatich kapitol a zaoberá sa výskumom alternatívnych spôsobov detekcie významných oblastí pre potreby ďalšieho spracovania obrazu.

**Kľúčové slová:** *významné oblasti, video, matlab*

## **Abstract**

Text anglickej verzie abstraktu

@TODO

**Keywords:** *saliency, video, matlab*

# Obsah

<b>1</b>	<b>Úvod</b>	<b>11</b>
<b>2</b>	<b>Prehľad literatúry</b>	<b>13</b>
2.1	Úvod do problematiky . . . . .	13
2.2	Metódy pre statické obrázky . . . . .	13
2.2.1	Baseline Center . . . . .	13
2.2.2	Hrany . . . . .	13
2.2.3	Ittiho model . . . . .	14
2.2.4	Spektrálne rezidua . . . . .	14
2.2.5	Sun Model . . . . .	15
2.2.6	Rare Model . . . . .	16
2.3	Metódy pre videá . . . . .	16
2.3.1	Zohľadnenie audio informácie . . . . .	17
2.3.2	Detekcia pohybu . . . . .	19
2.3.3	Lucas Kanade . . . . .	19
2.3.4	Horn-Schunck . . . . .	21
2.4	Metriky úspešnosti . . . . .	21
2.4.1	NSS . . . . .	22
2.4.2	AUC-Judd . . . . .	22
2.4.3	KL-Div . . . . .	22
2.5	Referčné datasety . . . . .	23
2.5.1	RSD . . . . .	23
2.5.2	SAVAM . . . . .	24
2.5.3	ASCMN dataset . . . . .	24
2.5.4	Coutrot dataset . . . . .	24
2.6	Porovnanie štandardných Metód . . . . .	25
<b>3</b>	<b>Špecifikácia</b>	<b>26</b>
3.1	Platforma pre riešenie . . . . .	26



3.2	Očakávané výsledky . . . . .	26
3.3	Ideálne Prípady . . . . .	26
3.4	Problémové Prípady . . . . .	26
<b>4</b>	<b>Implementácia</b>	<b>28</b>
4.1	Návrh metódy . . . . .	28
4.1.1	Dynamické príznaky videa . . . . .	28
4.1.1.1	Rozdiel smerových vektorov v horizontálnom smere . . .	29
4.1.1.2	Rozdiel smerových vektorov v vertikálnom smere . . .	29
4.1.1.3	Rozdiel vo vzdialenosti . . . . .	30
4.1.1.4	Spájanie regiónov . . . . .	30
4.1.1.5	Starnutie objektov na scéne . . . . .	31
4.1.2	Statické príznaky videa . . . . .	31
4.1.3	Výsledné spojenie príznakov . . . . .	31
4.1.4	Zdrojové kódy modelu . . . . .	32
4.1.5	Ukážky výsledkov . . . . .	32
4.1.5.1	Problémové typy videí . . . . .	33
4.1.6	Pipeline metódy . . . . .	35
4.2	Implementácia riešenia . . . . .	36
4.2.1	Aplikáciu na porovnávanie a automatickú validáciu . . . . .	36
4.2.1.1	Oddelenie logiky testovania a logiky samotného modelu .	36
4.2.1.2	Simultálne sledovanie videa z viacerých modelov . . . .	36
4.2.1.3	Automatická validácia modelu . . . . .	36
4.2.1.4	Vizualizácia výsledkov validácie . . . . .	37
4.2.2	Implementácia modulu . . . . .	37
4.3	Validácia výsledkov . . . . .	37
4.3.1	Analýza výsledkov . . . . .	38
4.3.1.1	ASCMN . . . . .	38
4.3.1.2	ASCMN - AUCROC . . . . .	38
4.3.1.3	ASCMN - KLDIV . . . . .	39
4.3.1.4	ASCMN - NSS . . . . .	39
4.3.1.5	Coutrot 1 . . . . .	39
4.3.1.6	Coutrot 1 - AUCROC . . . . .	39
4.3.1.7	Coutrot 1 - KLDIV . . . . .	39
4.3.1.8	Coutrot 1 - NSS . . . . .	40
4.3.1.9	Coutrot 2 . . . . .	40
4.3.1.10	Coutrot 2 - AUCROC . . . . .	40
4.3.1.11	Coutrot 2 - KLDIV . . . . .	40

4.3.1.12	Coutrot 2 - NSS . . . . .	40
4.3.1.13	Zhrnutie hodnotenia . . . . .	41
4.3.2	Porovnávanie s konkurenčnými modelmi pozornosti . . . . .	41
4.3.2.1	ASCMN . . . . .	41
4.3.2.2	Coutrot 1 . . . . .	44
4.3.2.3	Coutrot 2 . . . . .	46
4.3.2.4	Zhrnutie benchmarku . . . . .	48
4.3.3	Zhrnutie validacie . . . . .	48
4.4	Diskusia . . . . .	48
<b>5</b>	<b>Záver</b>	<b>50</b>
	<b>Zoznam použitej literatúry</b>	<b>54</b>

# 1. Úvod

Ľudské oko je schopné spracovať  $10^8$  až  $10^9$  bitov obrazových dát za sekundu. Ľudský mozog nie je schopný spracovať také množstvo dát naraz, preto sa získané informácie filtrujú pomocou ľudského vizuálneho systému[24]. Ľudský vizuálny systém je pravdepodobne najzložitejším mechanizmom akým človek disponuje. Je natoľko kľúčovým pre fungovanie spoločnosti či jedinca, že psychológovia sa zaoberajú jeho výskumom. Už viacero dekád študujú vlastnosti tohoto mechanizmu z pohľadu psychológie, fyziológie alebo neurobiológie.

Vyfiltrované oblasti obrazu už je možné spracovať vo výrazne rýchlejšom čase ako nefiltrované, ideálne v reálnom čase. Takéto oblasti sa nazývajú významné alebo charakteristické (v literatúre salient - prebrané z angličtiny). Významné oblasti sú vyberané pomocou mnohých faktorov. Najznámejšími sú prechody vo farbe, intenzite alebo orientácii. Ľudský vizuálny systém taktiež využíva skúsenosti pri pozorovaní. Oblasti takto vyfiltrované nesú pre pozorovateľa viac potencionálnych informácií ako ostatné oblasti obrazu a preto sa stávajú salientnými.

V systémoch počítačového videnia sa snažíme využívať primárne tieto oblasti pre pridelenie väčšej časti zdrojov. Z tohoto dôvodu je zistenie významných oblastí častým prvým krokom mnohých algoritmov v oblasti počítačového videnia.

Algoritmy na detekciu významných oblastí sa delia do troch skupín podľa princípu akým spracovávajú dáta[12]

1. Zdola-nahor: Prístup je cielený na nezávislosť od používateľa. Zameriava sa na fyziologicky významné oblasti vizuálneho systému ako výrazné zmeny v tvare, jase alebo farbe.
2. Zhora-nadol: Prístup je založený na čiastočnom riadení zo strany používateľa (konanie je podmienené úlohou). Riadenie je prínosom, pretože obsahuje aj informáciu používateľa a jeho predchádzajúcich vedomostí či skúseností, ktoré ovplyvňujú vnímanie.
3. Algoritmy využívajúce neurónové siete.

Cieľom práce je štúdium a výskum nových metód na detekciu významných oblastí vo videu. Následné porovnanie nových metód s existujúcimi v rôznych štandardných oblastiach.

**V prvej časti** sa nachádza prehľad metód na detekciu významných oblastí vo videu alebo metód na detekciu v statických obrazoch, ktoré majú potenciál pre použitie aj vo videu. Ďalej detailné vysvetlenie fungovania metód, ktoré budú použité v implementácii zlepšenia.

**V druhej časti** je popísaný postup a princíp zlepšenia a postup implementácie.

**Tretiu časť** tvorí validácia výsledkov a porovnanie výsledkov s inými modelmi, ktoré poskytujú lepšiu predstavu o efektivite algoritmu.

V závere....

## 2. Prehľad literatúry

### 2.1 Úvod do problematiky

Saliency a teda detekcia významných oblastí je využívaná v rôznych oblastiach. Automatizáciou sú modely významných oblastí (anglicky saliency modelov) ťažiskom pri segmentácii obrazu alebo detekcii špecifických objektov. Od saliency modelov sú taktiež závislé aj programy ovládajúce zabezpečovacie zariadenia. Tým, že zužujú možnosti a proaktívne upozorňujú na podozrivé situácie pomocou zúženia obrazu na zopár oblastí. Uplatnenie saliency model používa aj oblasť reklamy, kde je vizuálna pozornosť kľúčovým parametrom, čo môže rozhodnúť o úspechu produktu, veď aký význam by mala reklama, kde si nevšimnete prezentovaný produkt alebo si všimnete iba jeho 'menej' dokonalé časti.

### 2.2 Metódy pre statické obrázky

Algoritmy pre statické obrázky tvoria základ všetkých saliency modelov a tvoria najstaršiu oblasť výskumu. V tejto časti uvediem prehľad algoritmov pre výpočet saliency modelov od najjednoduchších cez najznámejšie až po najefektívnejšie. Na záver uvediem porovnanie všetkých metód pomocou všeobecne uznávaných metrík a dát získaných zo zariadení merajúcich pohyb očí používateľa (eyetrackera).

#### 2.2.1 Baseline Center

Baseline center je triviálny model, ktorý sa vypočítava pomocou Gaussovej krivky vzhľadom na pomer strán, čím predpokladá salientné oblasti presne v strede obrazu. Nezachytáva však žiadne sémantické aspekty videa, ako ani podvedomé informácie vnímanania obrazu, iba rozlíšenie dané optikou skenujúcou scénu.

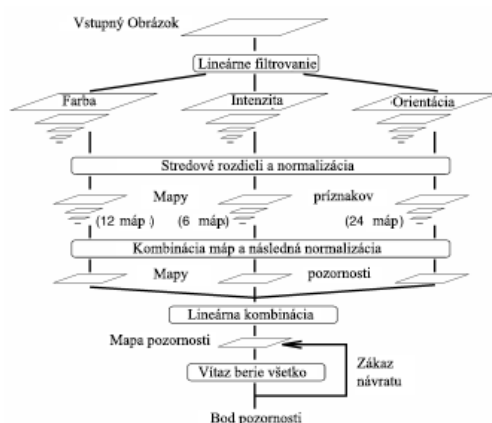
#### 2.2.2 Hrany

Skupinu algoritmov využívajúcu význačné prechody v obraze inak nazývame hrany. Metódy tohoto typu sú vyžívané hlavne v prírodných scénach, kde nie je (sémanticky) význačný objekt. Takéto metódy sa zakladajú priamo na štúdiu fyziologických vlastností ľudského

vizuálneho systému. Následná imitácia procesov odohrávajúcich sa na sietnici viedla ku vzniku saliency modelov generujúcich plausibilné výsledky[3].

### 2.2.3 Ittiho model

Najznámejším modelom pre výpočet významných oblastí pre statické farebné obrazy je ittiho model navrhnutý v roku 1998. Model zakladá na rozložení obrazu na tri základné charakteristiky obrazu a to farbu, intenzitu a orientáciu.

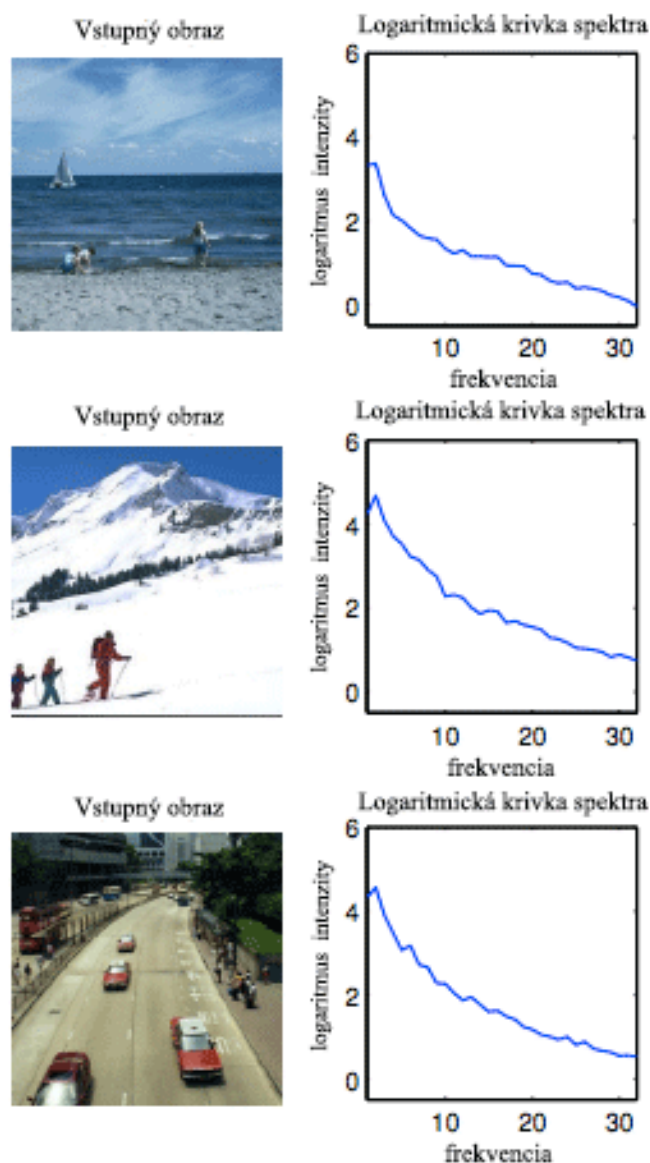


**Obr. 2.1:** Ucelená vizualizácia ittiho modelu[15]

Chrakteristika farby obsahuje 12 máp (šedotónové obrazy), pričom model používa farebný model RGB. Na začiatku sa vypočíta intenzita podľa vzťahu  $I = (R + G + B)/3$ . Pomocou mapy I sa následne normalizujú všetky farebné kanály modelu RGB. Model extrahuje štyri farebné kanály červený (r), zelený (g), modrý (b), žltý (y) a pomocou Gausových pyramíd vytvorí tri rôzne mapy každej farebnej zložky separátne. Červená zložka sa počíta difenčným spôsobom ako  $R = r - (g + b)/2$ , zelená ako  $G = g - (r + b)/2$ , modrá ako  $B = b - (r + g)/2$  a žltá ako  $Y = (r + g)/2 - |r - g|/2 - b$ . Chrakteristika intenzity obsahuje šesť máp. Získaná je pomocou orientovaných gáborových filtrov s orientáciou  $0^\circ, 45^\circ, 90^\circ, 135^\circ$ . Dokopy 42 máp charakteristík je následne lineárne skombinovaných do jednej saliency mapy[15].

### 2.2.4 Spektrálne rezidua

Metóda využíva princíp, že potláča štatisticky často opakujúce sa časti obrazu a do popredia stavia časti obrazu, ktoré sa štatisticky odlišujú od ostatných. Na detekciu používa rýchlu fourierovu transformáciu. Pomocou nej rozdelí obrázok na amplitúdovú časť a fázovú časť.



**Obr. 2.2:** Príklad rozloženia typovo rôznych obrázkov[14]

Amplitúdová zložka sa následne vyhladí, čím sa do popredia dostanú iba informácie, ktoré sa vymykajú z priemeru. Odčítaním od pôvodnej amplitúdovej zložky dostaneme iba časti obrazu, ktoré sú významné [14].

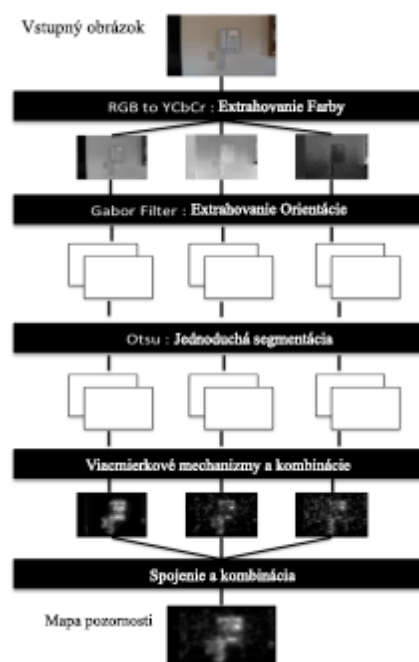
### 2.2.5 Sun Model

Sun model (Saliency Using Natural statistics) sa snaží simulovať potencionálne ciele sledovania ľudského vizuálneho systému. Model aktívne hodnotí tieto ciele odhadom pravdepodobnosti vzhľadom na všetky pozorované charakteristiky. Charakteristiky sú spracovávané separátne a teda model nepočíta s charakteristikami navzájom sa ovplyvňujúcimi. Údaje získané zo všetkých charakteristík následne spracuje štatisticky.

Model zakladá hlavne na Bayesovom pravidle. Za výsledok hľadania potom udáva asymetrie v týchto štatistických štruktúrach[25].

## 2.2.6 Rare Model

Výrazná väčšina modelov pozornosti typu bottom-up funguje ustáleným postupom, kde sa z pôvodného obrazu extrahuje definovaná množina chrakteristík paralelne a tie sa následne kombinujú alebo inak použijú na výpočet výslednej mapy pozonosti. Rare model narvhuje sekvenčnú architektúru, kde z pôvodného obrázku extrahuje nízko úrovňové príznaky. Následne na výsledkoch sériovo vykonáva extrakciu ďalších príznakov (v literatúre nazývané mid-level). Nakoniec ako posledný krok spojí a normalizuje výsledné chrakteristiky do konečnej mapy významných oblastí. Rare model ako nízkoúrovňové chrakteristiky používa jas a colorimetrické rozdiely (ako farebný model používa YCbCr) a následne na mapách rozložených žložiek farebného modelu detekuje orientáciu pomocou gáborových filtrov[21]. Po extrakcii všetkých chrakteristík použije iteratívnu metódu pre optimálne kvantovanie založenú na metóde Otsu[1]. Na takto upravenom vstupe sa následne vyhľadávajú vzácne (z angl. rare) oblasti obrazu. Metóda preskúmala možnosti nesekvenčnej extrakcie príznakov z obrazu čo bol novým prístupom v oblasti modelov pozornosti.



Obr. 2.3: Rare model workflow[21]

## 2.3 Metódy pre videá

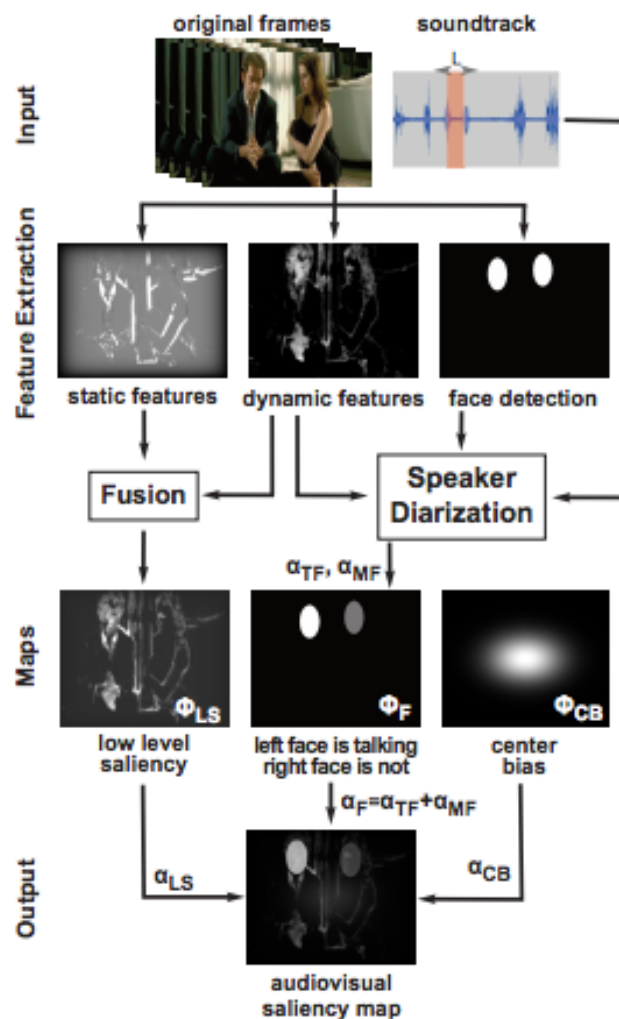
Video obsahuje rozsiahlejšie možnosti ako obrazová informácia, pribúdajú ďalšie rozmery ako je pohyb objektov na obraze alebo vplyv zvuku na ľudské vnímanie. Avšak oproti obrazu



je potrebné spracovávať väčšie množstvo dát. Navyše vo väčšine algoritmov využívajúcich saliency modely je potrebné, aby model dával výsledky v reálnom čase hlavne v oblasti zabezpečovacej techniky.

### **2.3.1 Zohľadnenie audio informácie**

Saliency modely využívajú rôznorodé druhy príznakov a to od geneticky zakorenených, ako sú prechody farieb, alebo intenzít, až po sémantické príznaky, ako je detekcia tváre [23]. Majoritná väčšina saliency modelov využíva iba obrazovú zložku, ale zvuková stopa býva nevyužívaná alebo úplne zanedbaná. Použitie zvuku je známym trikom filmovej scény už desaťročia. Režiséri posilňujú kontrolu nad diváckou pozornosťou práve pomocou zvukového doprovodu. Prvé štúdie sa zaoberali detekciou reči a tváre, kde je spojitosť jednoznačná [10]. Neskoršie štúdie dokazujú korelácie aj na všeobecnejšej úrovni a pokusy o extrakciu samotnej charakteristiky zo zvukovej stopy [11]. Tieto pokusy viedli aj k zostaveniu modelov zohľadňujúc zvukovú stopu ako samostatnú charakteristiku spolu s kombináciou nízkoúrovňových príznakov obrazu [6].



**Obr. 2.4:** Vizualizácia audiovizuálneho modulu[6]

Model extrahuje video na sekvenciu obrazov (framy) a audio stopu v tvare grafu vlnovej dĺžky. Potom extrahuje tri typy rôznych charakteristík. Nízkoúrovňové príznaky založené na biologicky inšpirovaných saliency modeloch rozdelených na dynamickú časť a statickú časť. Statická časť sa zameriava na najjasnejšie a najkontrastnejšie časti obrazu. Dynamická časť sa zameriava na relatívny pohyb objektov vzhľadom na pozadie (eliminácia pohybu kamery). Tieto 2 časti sa nakoniec spoja. Ďalšou charakteristikou použitou v tomto modeli je detekcia tváre. Každý objekt klasifikovaný ako tvár je v saliency mape nahradený oválnym objektom, intenzita daných objektov je daná pomocou metódy Speaker Diarization, ktorá detekuje podľa zvukovej stopy objekt ktorý generuje zvuk. Metóda predpokladá striedavú konverzáciu n objektov oddelenú pauzou. Následne spojí vyššie spomínané charakteristiky do jednej výslednej mapy. Ako posledný krok preloží cez celú mapu baseline center model popísaný v časti 2.2.1.

### 2.3.2 Detekcia pohybu

V tejto časti sa zmeriame na segmentáciu objektov ktoré sa na scéne pohybujú. Metódy tohoto typu sa snažia vizualizovať 3D prostredia (v našom prípade disponujeme výškou, šírkou, časom) na 2D výstup (obrazový výstup). Takáto informácia dokáže priblížiť výpočtové modely bližšie k realite. Ľudský vizuálny systém totiž nepoužíva iba 2d vstup (ako to prebieha vo drvivej väčšine metód na výpočet významných oblastí). Taktéto obrazy sú v ľudskom vizuálnom systéme vysoko hodnotené. Dôvody, prečo takto ľudský vizuálny systém pridáva prioritu práve takýmto oblastiam môžeme nájsť v antropológii (citácia?). Vysvetlenie je jednoduché a to snaha zabezpečiť bezpečné prostredie okolo seba a všetko pohybujúce sa narušuje pocit bezpečnosti. V nasledujúcom texte rozoberieme 2 najpoužívanejšie algoritmy používané na detekciu oblastí pohybu v obraze a výpočet vektoru posunu. Výpočet vektoru pohybu je však iba projekcia 3D vstupných dát do 2D obrazu, nemusí vždy reprezentovať iba pohyb. Prvým z nich bude LUCAS KANADE[4], a druhým Horn Schunck[2]. Oba tieto algoritmy používajú jeden spoločný predpoklad a to, že jas daného objektu sa časom nemení. To značí, že objekt sa na scéne môže presunúť ale svoj jas nemôže zmeniť. Matematicky vyjadrené  $I(x(t), y(t), t)$  je obrazová dvojrozmerná funkcia, ktorá sa mení vzhľadom na čas. Keďže sa jas obrazu nemení môžeme povedať, že platí:

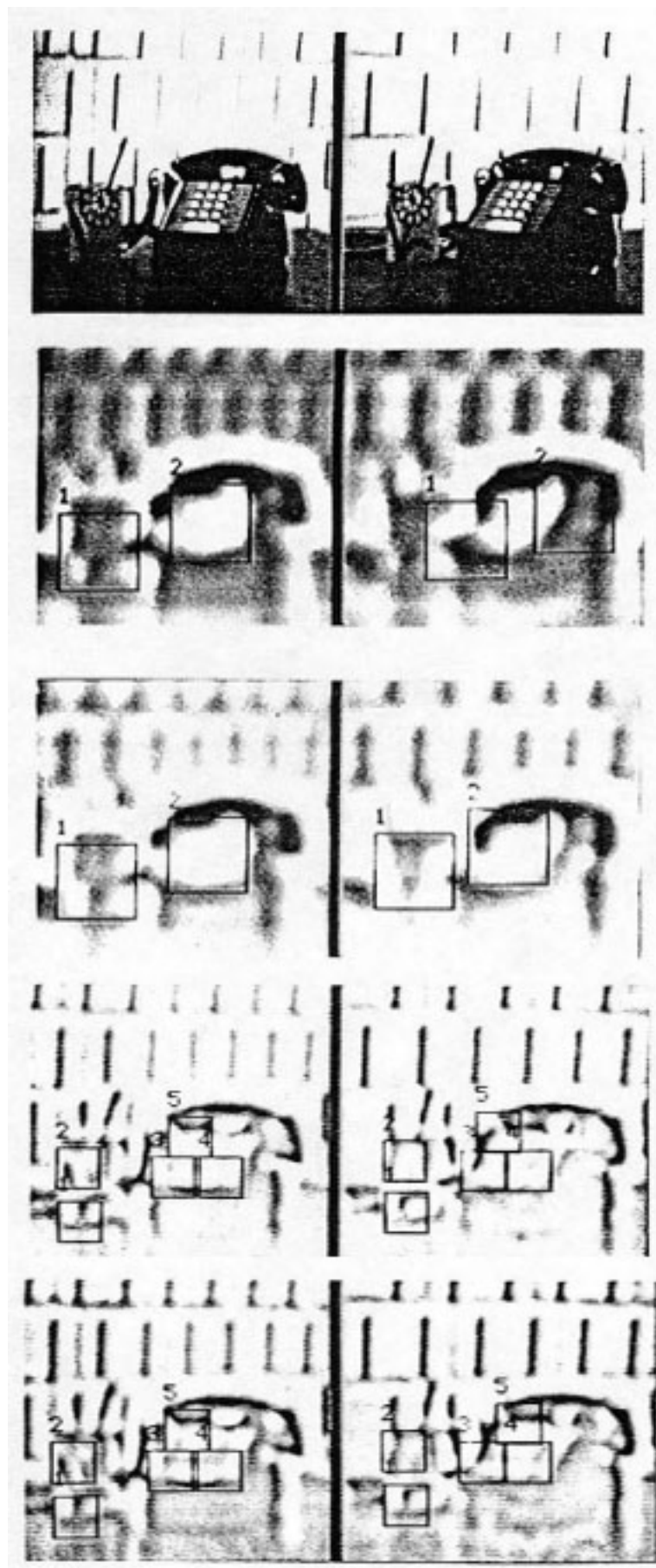
$$I(x + dx/dt, y + dy/dt, t + \Delta t) = I(x, y, t) \quad (2.1)$$

Z čoho je ľahko odvoditeľné, že:

$$dI/dt = dx/dt + dy/dt + dI/dt = 0 \quad (2.2)$$

### 2.3.3 Lucas Kanade

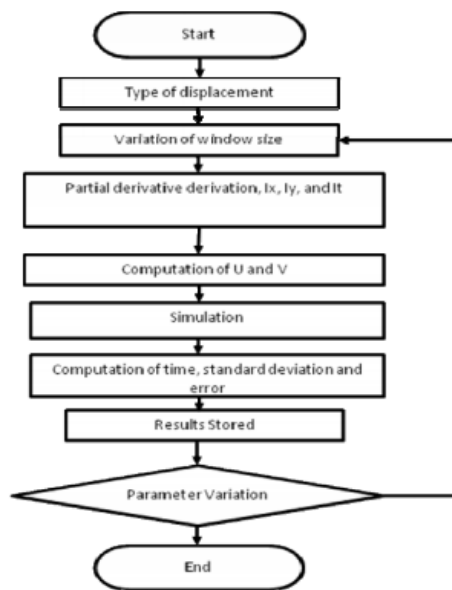
Algoritmus prvotne vznikol ako návrh pre časovú optimalizáciu problému výpočtu vektoru posunu medzi dvomi krivkami. Povodné intuitívne riešenie vyžadovalo  $O(M^2 * N^2)$  času pre výpočet daného vektoru ak M,N bolo rozlíšenie daného obrazového vzoru. Vtedy navrhovaná optimalizácia vyžadovala zadanie rozsahu hľadania, pomocou ktorého sa vypočítali diferencie pre celý obraz a pre ďalšiu iteráciu sa rozsah vypočítal pomocou horolezeckého algoritmu. Metóda Lucas Kanade využíva priestorový gradient pre výpočet nových hodnôt a zároveň upravuje hodnotu rozsahu pri výpočte každého obrazového pixelu v obraze a nie iba po výpočte celého obrazu. Pomocou takejto úpravy naivného algoritmu sa časová zložitosť zlepšila na  $O(M^2 \log N)$ [4].



**Obr. 2.5:** Vizualizácie výsledkov algoritmu Lucas-Kanade vždy po 1 iterácii[4]

### 2.3.4 Horn-Schunck

Metóda Horn-Schunck bola prvá, kde bola použitá metóda variácie na výpočet optického toku. Táto globálna metóda priniesla výpočet konštanty pre obmedzenie plynulosti optického toku. Algoritmus používa 2 základné parametre: Počet iterácií a vyhladzovaciu konštantu. Počet iterácií určuje dĺžku (počet cyklov) simulácie, vyhladzovacia konštanta je použitá po každom cykle simulácie kvôli zjemneniu prechodov a výpočet optimálneho optického toku.



Obr. 2.6: Vizualizácia pracovného postupu metódy Horn-Schunck

## 2.4 Metriky úspešnosti

Metriky úspešnosti sú algoritmy pre čo najpresnejšie vyjadrenie presnosti modelov v merateľných jednotkách. Takýto algoritmus dostáva na vstupe čisté dáta z eye trackera. Tieto je potrebné predspracovať z dôvodu, že každý výrobca poskytuje iné zariadenia na hardwarovej úrovni a výrobcovia neštandardizujú výstup do jednotnej formy. Následne je potrebné vytvoriť mapy fixácií ktorá sa používa ako jeden zo vstupných parametrov v algoritmoch rátajúcich metriky úspešnosti.

Metriky úspešnosti možno rozdeliť do 3 štandardných skupín podľa druhu hodnôt na ktorý porovnávajú reálne dáta (v literatúre nazívané ground truth) s vygenerovanými mapami význačných oblastí[20].

1. **Založené na porovnávaní hodnôt** - NSS, Percentile, Pf
2. **Založené na vyhodnocovaní vzdialeností** - AUC-Judd, AUC-Zhao, AUC Borji, AUC-Li

### 3. Založené na distribúcií - KL-Div, EMD, CC, SRCC

#### 2.4.1 NSS

NSS (Normalized Scanpath Saliency) metrika navrhnutá v roku 2005 ktorej autormi sú R. J. Peters a L. Itti. Metrika zakladá na ohodnotení salientných oblastí vzľadom na pozíciu fixácií samostatne a následná normalizácia podľa počtu fixácií.

Pre každú fixáciu používa vzťah

$$NSS(p) = (SM(p) - \mu_{SM}) / \sigma_{SM} \quad (2.3)$$

Kde SM je mapa význačných oblastí a p je bod danej fixácie pre ktorú sa hodnota vypočítava. Pričom mapa fixácií SM je normalizovaná tak aby nadobúdala nulovú strednú hodnotu a zároveň jednotkovú štandardnú odchýlku. Metrika NSS nadhodnocuje ak je saliency mape minimálna rozmanitosť hodnôt (malý rozdiel medzi hodnotami fixácií a strednou hodnotou), pretože v takomto prípade nebude model dostatočne odhodnotený, ak nájde presné pozície v prípade, že odchýlka je malá, alebo rozdiel medzi hodnotami fixácie a strednou hodnotou je vysoký. Finálna hodnota NSS metriky je určená priemerom hodnôt pre všetky fixácie[20].

$$NSS = 1/N * \sum_{p=1}^N NSS(p) \quad (2.4)$$

#### 2.4.2 AUC-Judd

Metrika je klasická AUC ktorú navrhol Judd [**auc-judd**]. Ako prvé sa pixely označené ako fixácie spočítajú s rovnakým počtom náhodných pixelov vybraných z mapy význačných oblastí a pixely sú nakoniec považované za klasifikátor úspešnosti. Následuje prahovanie zvolenou hodnotou, pixely ktoré sú menšie ako prahovacia hodnota sú pokladané za pozadie obrazu a pixely ktoré majú hodnotu vyššiu sú pokladané ako fixácie. Pre ľubovoľne zvolenú prahovaciu hodnotu sú niektoré výsledné oblasti manuálne označené ako pozitívne (True Positives), pobožne niektoré oblasti ktoré nie sú označené ako fixácie sú manuálne označené ako falošne pozitívne (False Positive). Tieto operácie sú zopakované tisíc krát, nakoniec sa vizualizuje pomocou ROC krivky a plocha pod pod krivkou (Area Under the Curve preto AUC) je výsledným klasifikátorom, ktorého ideálna hodnota je 1. Hodnota náhodného výberu je 0.5.

#### 2.4.3 KL-Div

V literatúre nazívaná aj Kullback-Leiblerova divergencia je bežne používaná aj mimo oblasti počítačového videnia, ako metóda pre odhad celkovej rozdielnosti medzi dvoma

distribúciami. Moho autorov saliency modelov používa túto metriku, ako hodnotu straty informácie tj. kolko informácií sa strati po vypočítaní daného saliency modelu voči mape fixácií.

Každý projekt vytvárajúci model významných oblastí si volí vlastné metriky úspešnosti, podľa ktorých sa určuje úspešnosť daného modelu. Pre meranie úspešnosti modelov je okrem samotných algoritmov pre meranie úspešnosti potrebné zabezpečiť dostatočne rôznorodú skupinu testovacích dát tkz. datasetov.

## 2.5 Referenčné datasety

Dataset je testovacia množina, ktorá sa snaží obsiahnuť dostatočne rôznorodé vzorky vhodné pre komplexné testovanie. Pri zostavovaní datasetov sú dôležité nielen videá ale eyetracker data alebo nejakým spôsobom zverejnené fixácie, aby bolo možné výsledky validovať pomocou vyššie uvedených metrík. Poslednou charakteristikou datasetu je množstvo ľudí na ktorých boli dané videá nahrávané.

Príklady datasetov:

- **RSD**[16]
- **SAVAM**[13]
- **Coutrot datasets**[8]
- **ASCMN**[22]

### 2.5.1 RSD

Regional Saliency Dataset je zaujímavý o čo najobširnejšie testovanie je rozdelený do 4 hlavných kategórií:

- **bezpečnostné záznamy** - Štandardné záznamy z bezpečnostných kamier obsahujú statické pozadie a salientné pohybujúce sa objekty. Pre túto časť datasetu vyžili záznamy z projektu CAVIAR[18].
- **Grafika** - Použité animované filmy/seriáli ktoré obsahujú 2D aj 3D grafiku.
- **Prirodzené videá s prvkami grafiky** - Prirodzené videá podobné bezpečnostným ale s prvkami umelo vložených priamo do obrazového kanálu.
- **Prirodzené videá** - Videá bez pridaných grafických prvkov, tak ako boli nasnímané kamerou.

Na vyznačenie zaujímavých oblastí nezvolili techniku (eyetracke) ale manuálne vyznačovanie zaujímavých oblastí pomocou používateľov. Výskumu sa zúčastnilo 17 mužov 6 žien medzi 10-23 rokov, na označení každého z videa sa podieľalo 10-23 ľudí.



**Obr. 2.7:** Ukážka z každej kategórie videa s oznčenými významnými областami

## 2.5.2 SAVAM

SAVAM (Semiautomatic Visual-Attention Modeling) je dataset nahrávaný priamo pomocou eyetrackera pri sledovaní videí v HD rozlíšení pričom každému nahrávanému používateľovi sú pridelené dáta separátne pre každé oko. Dokopy obsahuje 13minút videa, ktoré bolo otestované na 50 používateľoch rôzneho veku. Dataset je rozdelený na videá z filmov, ukážky z komerčných videí a stereoskopické videá. SAVAM taktiež poskytuje všetky raw dáta z eyetrackera ako aj vizualizácie daných dát[13].

## 2.5.3 ASCMN dataset

ASCMN nazvaný podľa rozčlenenia do 5 skupín videa: Abnormálne, bezpečnostné, videá s davom, videá s pohybom a videá s chybami v obraze (z anglických názvov: Abnormal, Surveillance, Crowd, Moving, Noise). Spolu obsahuje 24 videj, každé video bolo namerané na 10 rôznych používateľoch. K datasetu je taktiež dostupný validačný kód[22] vypočítavajúci hodnotiace metriky na ľubovoľnom modeli pozornosti.

## 2.5.4 Coutrot dataset

Ide o 2 rôzne datasety, oba sú nazvané podľa jeho autora, Antoine Coutrota. Prvý dataset[10] obsahuje videá s dynamickou povahou scény. Je rozčlenený do 4 vizuálne rozličných kategórií.

- Jeden pohybujúci sa objekt
- Viac pohybujuúcich sa objektov
- Prírodné scény
- Konverzačné scény



Tento dataset obsahuje dokopy 60 videí, ktoré sledovalo vždy 18 rôznych používateľov. Všetky videá boli zaznamenávané v 4 rôznych zvukových podmienkach (využívať budeme iba dáta s pôvodnou zvukovou stopou).

Druhý dataset [7] obsahuje 15 videí. Všetky videá obsahujú nahraté stretnutie 4 konverzujúcich ľudí so statickou kamerou, dataset nieje členený do žiadnych kategórií. Dáta oboch datasetov boli nahrávané pomocou eyetrackera EyeLink 1000 pri 1000Hz, pričom používatelia sedeli 57cm od monitoru. Eyetracker nahrával iba dáta z dominantného oka pre daného používateľa.

## **2.6 Porovnanie štandardných Metód**

Porovnávanie metód je štandardne publikované formou ucelených benchmarkov. Príkladom takéhoto banchmarku je mit saliency benchmark[5], ktorý sa snaží zgrupovať a porovnávať obrazové modely pozornosti a zverejňovať referencie na ďalšie podobné projekty. Pre účely validácie bude vypracovaný podobný benchmark určený pre porovnanie rôznych modelov pozornosti na typovo rôznych datasetoch.

## 3. Špecifikácia

### 3.1 Platforma pre riešenie

Ako platforma pre implementáciu budeme používať programovací jazyk Matlab. Všetky zdrojové kódy budu poskytnuté výhradne v tomto vývojovjárskom jazyku.

### 3.2 Očakávané výsledky

Výsledkom práce bude model pozornosti, ktorý zohľadňuje príznaky extrahovateľné iba z videa a nie z čisto obrazovej informácie. Pôjde hlavne o pohyb objektov na scéne a iné sémantické informácie, ktorými sa video odlišuje od statickej scény. Príkladom nového rozmeru videa okrem možnosti pohybu objektov, ktorú musíme počítať pri tvorbe mapy pozornosti. Sekundárnym prínosom práce bude vytvorenie jednotnej aplikácie pre vizuálne porovnávanie modelov, kde používateľ bude môcť jednoducho pridávať modely, ideálne priamo použiť ukážkové zdrojové kódy zverejnené autormi jednotlivých modelov alebo úpravou, ktorá nevyžaduje znalosť logiky stojacej za daným modelom. Následne automatický výpočet štandardných metrík na implementovanom datasete, pre jednoduchú validáciu výsledkov na rovnakých dátach, spolu s konkurečnými modelmi z dôvodu jednoduchého ladenia počas vývoja modelu.

### 3.3 Ideálne Prípady

Idálne prípady očakávam v prípade použitia záznamov z bezpečnostných kamier, z dôvodu statickej kamery. Vďaka statickému pozadiu sú výsledky detekcie optického toku objektov najrelevantnejšie. To predurčuje takého videá k najlepším výsledkom.

### 3.4 Problémové Prípady

Najproblémovejšími vstupmi sú očakávané videá s dynamickým pohybom kamery kombinovaným s pohybom objektov alebo videá s fixovanou kamerou na objekt dynamicky sa pohybujúci sa po scéne. Vo videách takéhoto charakteru predpokladám chybné označovanie oblastí a z toho vyplývajúce chyby v mapách pozornosti, preto sa budem snažiť v týchto

prípadoch utlmovať dynamické príznaky videa a používať iba statické príznaky aktuálneho obrazu.

## 4. Implementácia

### 4.1 Návrh metódy

Navrhovaná metóda zohľadňuje vlastnosti, ktoré nie je možné získať iba zo statického obrazu, budeme ich nazývať dynamické príznaky videa. Avšak metóda stále zohľadňuje v pozorovanom videu aj aspekty statického obrazu, tieto budeme nazývať statické príznaky videa. Tieto príznaky sú vypočítavané separátne a nakoniec ich metóda spája do jednej výslednej mapy pozornosti. Výsledkom je postupnosť máp pozornosti pre každý frame videa (podľa vstupnej konfigurácie), ktorý možno spojiť do videa pozornosti pre ľubovoľné vstupné video.

#### 4.1.1 Dynamické príznaky videa

Dynamické príznaky metóda najprv extrahuje pomocou štandardnej metódy Horn-Schunck, (referencia na 2 kapitolu alebo na článok?) ktorá vypočíta optický tok na každých 2 rozdielnych framoch videa, čím vzniká sémantický príznak pohybu rôznych objektov po scéne spolu s smerovými vektormi pohybu daných vektorov. Získané smerové vektory okamžite spočítavame, aby sme získali celkový obraz optického toku pre danú dvojicu obrazov. Obraz sa následne prahuje statickou konštantou kvôly odstráneniu šumu. Prahovanie prebieha dynamicky vzhľadom na počet nájdených 8-spojitéch regiónov tj. výstup optického toku. V našej implementácii je obmedzený počet regiónov na maximálnu hodnotu 200 regiónov. Prahovanie začne s konštantou, ktorú určí pomocou algoritmu Otsu[1], následne určí počet 8-spojitéch regiónov. Ak je počet väčší ako maximálna hodnota, zvýši konštantu o 10. Tento proces sa opakuje pokiaľ sa v obraze vyskytuje viac ako maximálny počet regiónov. Takéto prahovanie je nutné pre optimalizáciu výkonu algoritmu, pretože v prípadoch keď obraz obsahuje veľké množstvo regiónov, výpočtová rýchlosť algoritmu je maximálne neúčinná. Pixely s validnou hodnotou sa rozdelia na regióny podľa spojitosti a podobnosti štandardným spôsobom. Pripomenme, že v tomto obraze sa spočítali hodnoty posunu v oboch smeroch aritmeticky do jednej hodnotiacej konštanty (pre každý pixel obrazu), ktorá už nereprezentuje smer posunu daného obrazového pixelu, ale iba hodnotí celkový posun pixelu. Takto získané regióny budeme vyhodnocovať a spájať podľa pôvodných výsledkov metódy Horn-Schunck. Vďaka využitiu pôvodných vektorov z výsledku metódy

Horn-Schunck, vieme rozlíšiť pohyb horizontálny aj vertikálny separátne. Pre všetky dvojice regiónov v obraze zisťujeme nasledovné charakteristiky:

1. **Rozdiel smerových vektorov v horizontálnom smere**
2. **Rozdiel smerových vektorov v vertikálnom smere**
3. **Rozdiel vo vzdialenosti**

#### **4.1.1.1 Rozdiel smerových vektorov v horizontálnom smere**

Charakteristika sa vypočítava zo smerových horizontálnych vektorov metódy Horn-Schunck. Pre každý región sa vypočíta maximálna hodnota z indexov daného regiónu. Následne sa za hodnotu charakteristiky sa považuje absolútna hodnota rozdielu týchto hodnôt pre každý región.

$$H_A = \max(HS(i_A)) \quad (4.1)$$

$$H_B = \max(HS(i_B)) \quad (4.2)$$

$$R_H = \text{abs}(H_A - H_B) \quad (4.3)$$

Kde A, B reprezentujú všetky dvojice regiónov, ktoré sa nachádzajú v obraze.  $V_a, V_b$  je maximálna hodnota horizontálnych smerových vektorov z výsledku Horn-Schunck algoritmu pre všetky oblasti patriace danému regiónu.  $R_H$  je výsledná hodnota charakteristiky.

#### **4.1.1.2 Rozdiel smerových vektorov v vertikálnom smere**

Charakteristika sa vypočítava zo smerových vertikálnych vektorov metódy Horn-Schunck. Pre každý región sa vypočíta maximálna hodnota z indexov daného regiónu. Následne sa za hodnotu charakteristiky považuje absolútna hodnota rozdielu týchto hodnôt.

$$H_A = \max(HS(i_A)) \quad (4.4)$$

$$H_B = \max(HS(i_B)) \quad (4.5)$$

$$R_V = \text{abs}(H_A - H_B) \quad (4.6)$$

Kde A, B reprezentujú všetky dvojice regiónov ktoré sa nachádzajú v obraze.  $V_a, V_b$  je maximálna hodnota vertikálnych smerových vektorov z výsledku Horn-Schunck algoritmu pre všetky oblasti patriace danému regiónu.  $R_V$  je výsledná hodnota charakteristiky.

#### 4.1.1.3 Rozdiel vo vzdialenosti

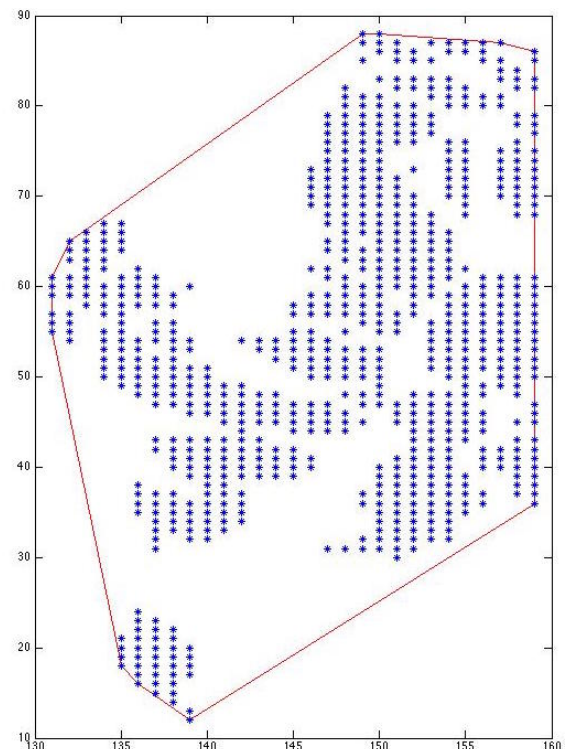
Chrakteristika sa vypočítava ako minimálna hodnota vzdialenosti medzi dvojicou regiónov. Hodnota je počítaná euklidovskou metódou.

```
forall the rohA ako každý extrém regiónu A do
| forall the rohB ako každý extrém regiónu b do
| | vzdialenosť = sqrt( (corner2(1,1)-rohB(1,1))2 + (rohB(1, 2) - rohA(1, 2))2)
| end
end
```

**Algorithm 1:** Výpočet minimálnej vzdialenosti euklidovskou metódou

#### 4.1.1.4 Spájanie regiónov

Po výpočte všetkých 3 charakteristík spojíme všetky dvojice regiónov, pre ktoré sú všetky charakteristiky nižšie ako zadaná konštanta. Regióny spájame pomocou konvexného obalu zjednotenia bodov ležiacich v oboch regiónoch.



**Obr. 4.1:** Vizualizácia spojenia regiónov pomocou konvexného obalu

#### 4.1.1.5 Starnutie objektov na scéne

Do vypočítavania dynamických príznakov započítavame predpoklad, že aj pohybujúce sa objekty postupne strácajú pozornosť používateľov. A to v prípade kedy sa síce daný objekt na scéne pohybuje, ale na identickom mieste. Do metóty zabudujeme mechanizmus, kde pixelom s dlhodobou vysokým hodnotením pozornosti, zmenšíme toto hodnotenie pomocou vynásobenia koeficientom hodnoty 0 to 1.

#### 4.1.2 Statické príznaky videa

Pri videách, kde sa pohybuje celá scéna (kamera je v pohybe) nedávajú dynamické príznaky dobré výsledky keďže logicky označia celú scénu alebo väčšinovú časť scény za výrazne salientnú. Preto je vhodné dynamické príznaky vhodne kombinovať s klasickými modelmi pozornosti, ktoré síce zanedbávajú postupnosť obrazov, ale nezlyhajú ako dynamické príznaky. Pre extrakciu statických obrázkov sme zvolili metódu založenú na spektrálnych reziduách[14]. Vďaka svojmu princípu potlačovania štatisticky opakujúcich sa predmetov na scéne, sa dá predpokladať vhodné doplnenie statických objektov, ktoré môžu zaujať pozornosť na videu ak zlyhávajú dynamické príznaky.

#### 4.1.3 Výsledné spojenie príznakov

Spájanie dynamických a statických príznakov bude prebiehať pomocou sčítania oboch máp, pričom vždy sa použijú v určitom pomere. Výpočet pomeru bude určovať pomer výskytu salientných pixelov v mape dynamických príznakov.

$$pomer = \sum_{n=0}^{Pix_{count}} P_D(n) > 0 / Pix_{count} \quad (4.7)$$

Kde  $P_D$  reprezentuje mapu dynamických príznakov a  $Pix_{count}$  je počet všetkých pixelov, ktoré obraz obsahuje.

Ak je vysoký výskyt salientných pixelov, potrebujeme utlmiť zobrazovanie tejto časti príznakov a prioritizovať zobrazovanie statických príznakov, preto zmiešavacia funkcia vyzerá nasledovne:

$$Výsledok = (P_D * (1 - pomer)) + (P_S * pomer) \quad (4.8)$$

Kde  $P_D$  reprezentuje mapu dynamických príznakov a  $P_S$  mapu statických príznakov.

V prípade, že algoritmus nedokáže detekovať žiadny pohyb na scéne, bol by model pozornosti prázdny. Preto v prípade, keď je vyššie spomýnaný pomer dynamických pixelov extrémne nízky použijeme ako výstup algoritmu iba statické príznaky. Naopak v prípade,

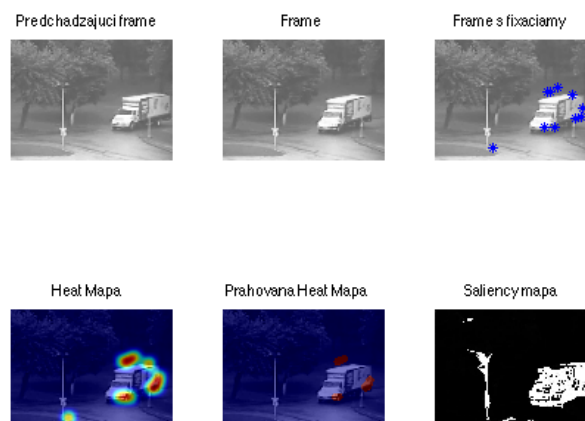
že kamera je v pohybe Horn-Schunck algoritmus označí ako pohybujúci sa väčšinou oblasť obrazu a v tom prípade je potrebné utlmiť dynamické príznaky obrazu a do popredia vystupujú statické.

#### 4.1.4 Zdrojové kódy modelu

Zdrojový kód obsahuje jednu metódu, ktorá prijíma na vstupe vždy 2 parametre. Prvý parameter je aktuálny frame videa a druhý parameter je frame videa určený na extrakciu dynamických príznakov videa pomocou difference vzľadom na prvý obrazový frame. Tieto 2 obrazové vstupy nemusia byť nutne po sebe idúce, je na používateľovi či použije model serializovane na každý frame videa, alebo zvolí vlastnú implementáciu keyframingu (napríklad kvôly časovej náročnosti algoritmu). Algoritmus je schopný processovať farebné aj čiernobiele obrazové vstupy. V prílohe je možné nájsť 2 implementácie a to implementáciu modelu pre aplikáciu na porovnávanie modelov načítavajúca každé 2 posebeidúce obrazové framy. Druhá implementácia načítava na vstupe priamo video a na výstup dá video s korešpondujúcim videom význačných oblastí, táto implementácia je určená na použitie mimo aplikácie na testovanie. Obe implementácie sú dostupné v prílohe na CD alebo voľne dostupné na internete.

#### 4.1.5 Ukážky výsledkov

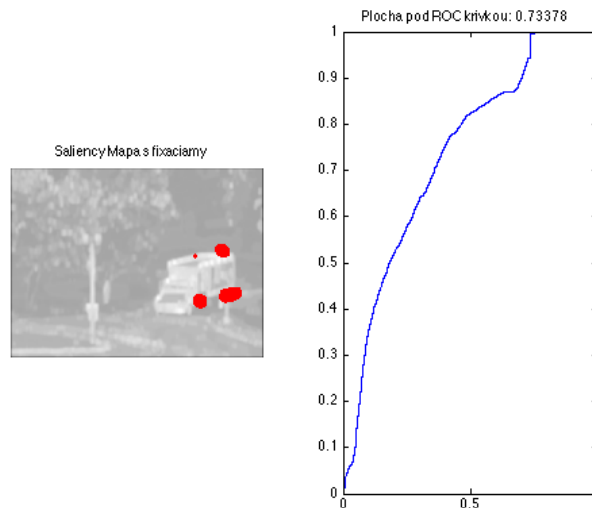
V tejto sekcii budem prezentovať výsledky konkrétne prípady videí (framy) zachytené počas testovania a validácie. Vždy uvediem po sebe nasledujúce originálne framy videa a vizualizáciu fixácií. Ako príklad uvediem frame 53 z videa č. 23 z datasetu ASCMN[22].



**Obr. 4.2:** Porovnanie výstupu mapy pozornosti a reálnych dát



Grafy sú generované pomocou pozmeneného ukážkového skriptu distribuovaného spolu s datasetom ASCMN[22] a porovnávané s výsledkom navrhovaného modelu pomocou grafu 4.2. V prvom riadku vidíme ako prvý pôdovdný frame (n-1) nasledovaný testovaným framom posledný obrázok zobrazuje fixácie. V druhom riadku uvádzam postupne heatmapu pre daný frame, prahovanú heat mapu a ako poslednú výslednú saliency mapu.

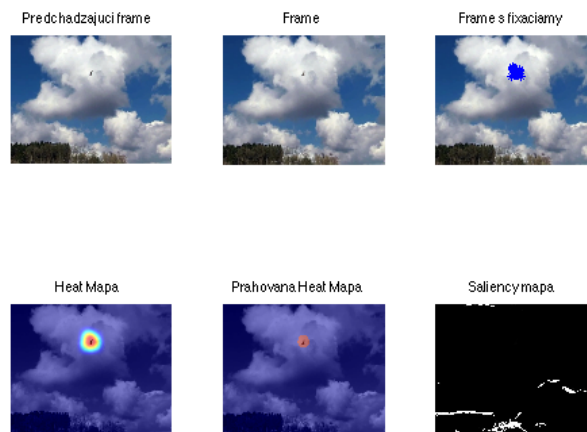


**Obr. 4.3:** Vizualizácie metriky AUC-Judd pomocou kódu zverejneného v mit saliency benchmark[5]

Z grafov 4.2 vidno koreláciu dát, čo taktiež povrdzuje metrika vypočítaná na danom frame na grafe 4.3 reprezentujúca AUC-Judd[**auc-judd**] krivku.

#### 4.1.5.1 Problémové typy videí

V tejto sekcii uvediem typové video s rovnakou analýzou ako je uvedená vyššie, iba na typ videa bude nevhodný navrhovaný model. Typovo videá možno označiť ako videá s fixovanou kamerou v pohybe. Tj. kamera je fixovaná k sledovanému objektu na scéne, z čoho vyplýva, pozadie obrazu (okolie sledovaného objektu) je pre náš algoritmus v pohybe aj keď reálne sa hýbe kamera a preto je považované za významný objekt. Zároveň sledovaný objekt (často aj objekt v pozornosti používateľov) sa vizuálne nepohybuje, z dôvodu, že jeho pohyb je kompenzovaný fixáciou kamery a preto ho navrhovaný algoritmus považuje za vizuálne nevýznamný. Ako príklad uvediem frame č.80 z videa č. 10 z datasetu coutrot 1.

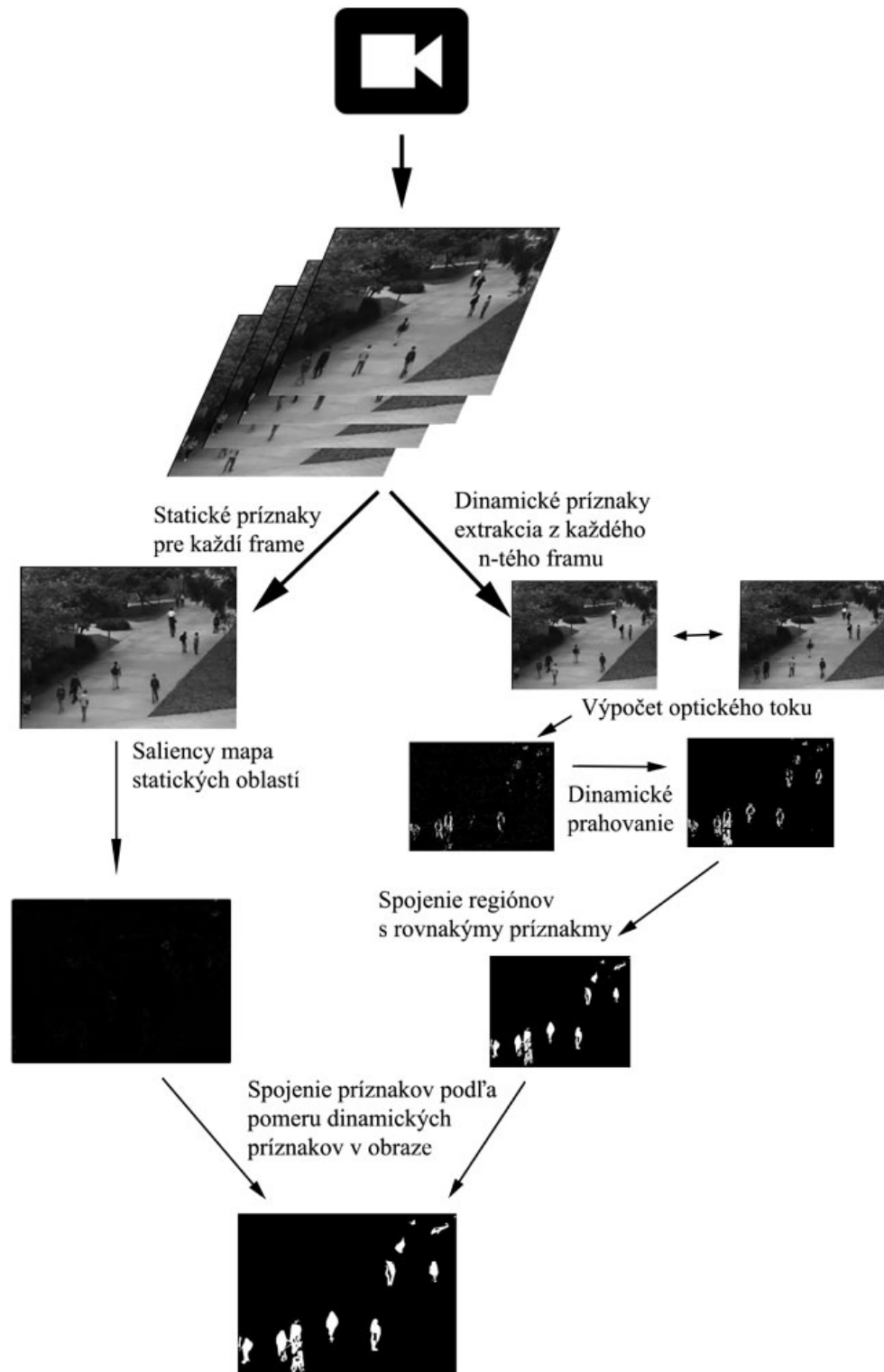


**Obr. 4.4:** Porovnanie výstupu mapy pozornosti a reálnych dát

Už podľa grafu 4.1.5.1 nemá výsledok navrhovaného modelu žiadnu koreláciu s reálnymi dátami. Preto nebudeme ďalej uvádzať žiadnu metriku. Návrh riešenia pre tieto prípady budú rozanalyzované v sekcii 4.4.

#### 4.1.6 Pipeline metódy

Grafický popis metódy obsahujúci graf zostavenia mapy pozornosti.



Obr. 4.5: Ucelená vizualizácia algoritmu

## 4.2 Implementácia riešenia

Implementácia vyššie uvedeného algoritmu je implementovaná ako modul pre aplikáciu na porovnávanie a automatickú validáciu výsledkov. Aplikácia na porovnávanie je takisto implementovaná v prostredí matlab.

### 4.2.1 Aplikáciu na porovnávanie a automatickú validáciu

Sekundárnym prínosom práce je vytvorenie aplikácie pre zjednodušenie budúcej práce pri prototypovaní nových modelov pozornosti. A následné uľahčenie validačného procesu pre potencionálnych vývojárov.

Základná functionalita:

1. **Oddelenie logiky testovania a logiky samotného modelu**
2. **Simultálne sledovanie videa z viacerých modelov**
3. **Automatická validácia modelu**
4. **Vizualizácia výsledkov validácie**

#### 4.2.1.1 Oddelenie logiky testovania a logiky samotného modelu

V aplikácii na testovanie je možné pridávať ľubovoľné modely, pre ktoré je dostupná implementácia v jazyku matlab. Pre iné jazyky je potrebné doprogramovať wrapper, ktorý spustí daný jazyk a vypočíta mapu pozornosti. Ukážkový wrapper je súčasťou aplikácie. Pre pridanie nového modelu je potrebné pridať wrapper do zložky "models", knižnice vyžadované modelmi je potrebné skopírovať do ľubovoľnej podzložky tohoto priečinka. Pri spustení aplikácie sa načítajú všetky moduly aj knižnice uložené v podzložkách.

#### 4.2.1.2 Simultálne sledovanie videa z viacerých modelov

Pre rýchle prototypovanie je vhodné pozorovať rovnaké video pri rôznych úpravách. Táto funkcionalita je dostupná pre každý model s vygenerovanými mapami pozornosti na zvolenom videu.

#### 4.2.1.3 Automatická validácia modelu

Validovanie výsledkov je nutnou súčasťou každého modelu pozornosti preto aplikácia ponúka automatizovaný spôsob ako zvalidovať výsledky na vybraných referenčných datasetoch. Validácia tvorí pre každé video perzistentný súbor obsahujúci 3 metriky: AUC-Judd, KL-Div, NSS. Vyššie spomenuté metriky sa rátajú pre každý frame videa.

Validácia prebieha paralelne pre všetky videá zvoleného datasetu. Vytvorené súbory sú perzistentné z dôvodu dlhého výpočtového času a ukladajú sa do priečinku `results` a podložky podľa názvu testovaného datasetu v tvare *názovModelu.názovDatasetuČísloVidea.mat*. Formát súboru obsahuje 3 premenné s názvami: `AUROC_score`, `KLDIV_score`, `NSS_score`. Každá premenná obsahuje pole podľa dĺžky videa (počet frameov) a hodnotami danej metriky. Aplikácia aktuálne podporuje 2 datasety a to: `ASCMN`[22], `Coutrot` testovacie datasety 1[9] a 2[7], tieto datasety sú voľne dostupné a súčasťou aplikácie je programový kód, slúžiaci na načítanie a validovanie výsledkov (samotné vstupné videá a fixácie je potrebné stiahnuť zo stránky autorov) Dataset `ASCMN`[22] je poskytovaný autormi aj s testovacím algoritmom na výpočet vyššie uvedených metrík, do aplikácie na testovanie bol iba pozmenený pre načítanie ľubovlného modelu a prispôbený na paralelný výpočet všetkých videí paralelne. Pre `Coutrot` datasety aplikácia na testovanie obsahuje upravenú verziu validačného algoritmu z datasetu `ASCMN`.

#### 4.2.1.4 Vizualizácia výsledkov validácie

Pre analýzu výsledkov validácie dokáže aplikácia prehľadne vizualizovať všetky dáta získané testovaním. Vizualizácie sú súčasťou validácie v ďalších kapitolách.

#### 4.2.2 Implementácia modulu

Implementácia nového modelu pozornosti je jednoduchá. Pre integrovanie ľubovlného modelu je možné použiť vzorovú implementáciu, ktorá je dostupná v prílohách.

### 4.3 Validácia výsledkov

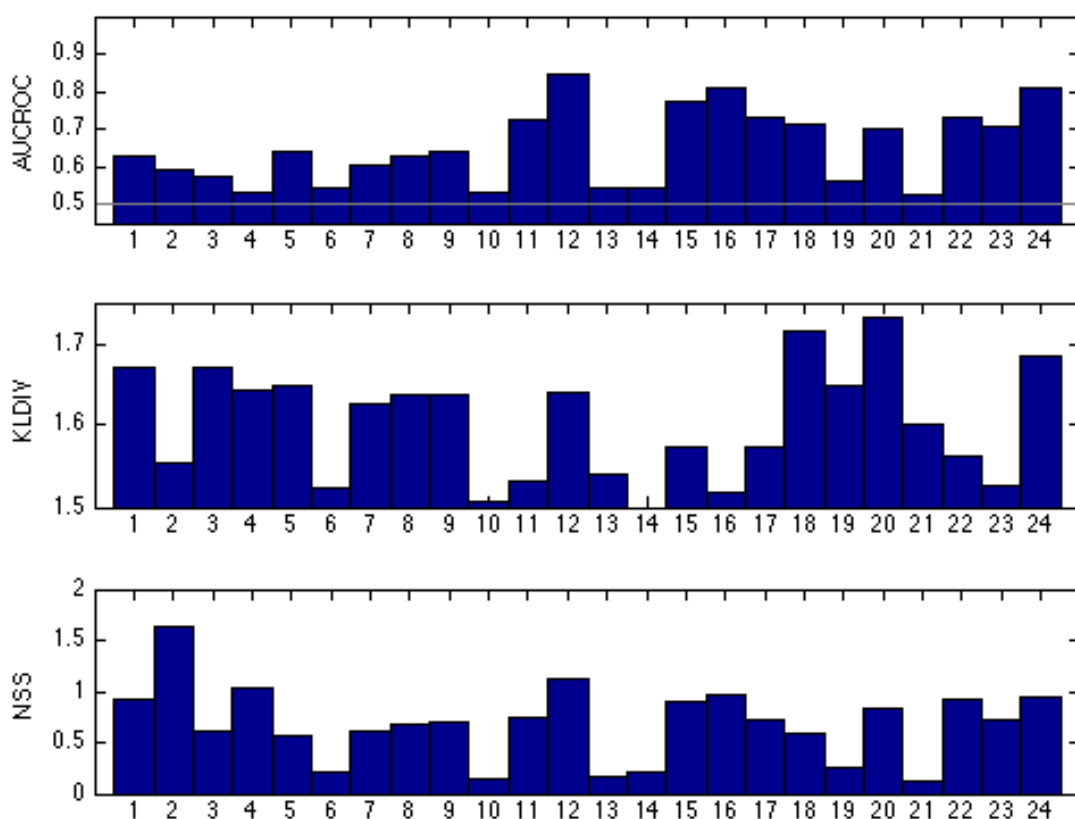
Validácia vyššie spomínaného modelu prebiehala pomocou automatického testovania v aplikácií na testovanie. Prezentovať budem výsledky z nasledujúcich datasetov: **`ASCMN`**[22], **`Coutrot 1`**[9], **`Coutrot 2`**[7]. Výsledky budem hodnotiť pomocou nasledujúcich metrík: **`AUCROC`**[20], **`KLDIV`**[20], **`NSS`**[20]. V nasledujúcich sekciách budem prezentovať výsledky validácie pre navrhovaný model a ďalej vyhodnocovať vytvorený benchmark.

@TODO screens+postup of validation???

### 4.3.1 Analýza výsledkov

V tejto sekcii budem prezentovať výsledky všetkých datasetov vzhľadom na navrhovaný model. Výsledky budem vizualizovať pomocou charakteristiky vzniknutej zo strednej hodnoty framov, jednotlivých videí. Každá metrika bude vyhodnocovaná samostatne. Ako prvý budeme analyzovať dataset ASCMN[22] a následne oba Coutrotove datasety[9] [7].

#### 4.3.1.1 ASCMN



**Obr. 4.6:** Vizualizácia všetkých testovaných metrík pre dataset ASCMN[22] pre jednotlivé videá

#### 4.3.1.2 ASCMN - AUCROC

Ideálna hodnota tejto metriky je 1, čo reálne značí 100% úspešnosť. Náhodná mapa pozornosti má hodnotu 0.5, z toho dôvodu aby sme dokázali správnosť nového modelu potrebujeme dokázať, že model má hodnotu AUC v intervale  $[0.5, 1]$ . Z grafu 4.6 vyplýva, že všetky videá spĺňajú vyššie uvedenú podmienku. Výsledná priemerná hodnota pre AUC 0.6515. Táto hodnota už priamo dokazuje túto hypotézu, keďže ho tvorí stredná hodnota všetkých testovaných videí. Keďže je hodnota dostatočne vyššia ako 0.5, hodnota vyhodnocuje náš model ako korelujúci s reálne nameranými dátami na používateľoch zúčastnených sa na tvorbe tohoto datasetu.

#### 4.3.1.3 ASCMN - KLDIV

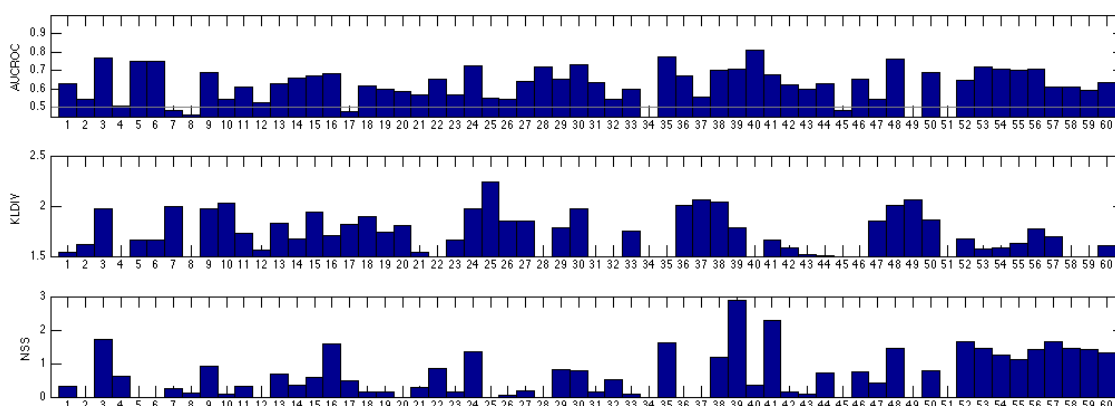
Ideálna hodnota tejto metriky je 0, čo reálne značí že saliency mapa je totožná s ground truth mapou. Výsledné hodnoty pre všetky videá sa podľa grafu 4.6 pohybujú v intervale  $[1.5, 1.7]$ . Hodnoty menšie ako 2 značia tiež koreláciu s reálnymi dátami. Výsledná hodnota KLDIV je 1.6027 je nižšia ako 2, preto aj metrika KLDIV úspešne validuje náš model vrámci datasetu.

#### 4.3.1.4 ASCMN - NSS

Výsledná hodnota NSS je 0.6809. Hodnota NSS metriky odhaluje, že hodnoty v oblastiach fixácií minimálne s normalizovanými fixáciami a model by mal zmeniť hodnoty v oblastiach fixácií.

#### 4.3.1.5 Coutrot 1

V datasete Coutrot 1 boli z testovania vynaté videá číslo: 10, 15 z dôvodu chybného zdrojového videa alebo fixácií zverejnených jeho autormy.



**Obr. 4.7:** Vizualizácia všetkých testovaných metrík pre dataset Coutrot 1[9] pre jednotlivé videá

#### 4.3.1.6 Coutrot 1 - AUCROC

Výsledná priemerná hodnota pre AUC 0.6086. HTakáto hodnota vyhodnocuje náš model ako korelujúci s reálne nameranými dátami na používateľoch zúčastnených sa na tvorbe tohoto datasetu.

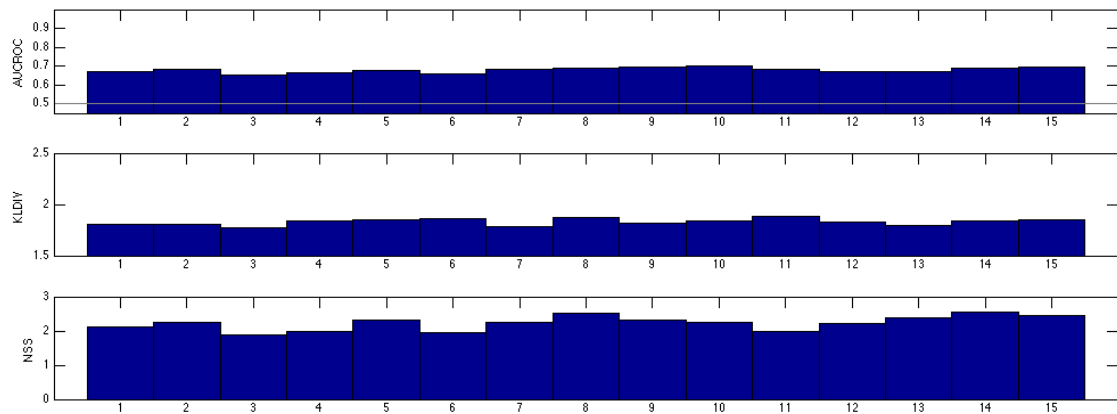
#### 4.3.1.7 Coutrot 1 - KLDIV

Výsledná hodnota KLDIV je 1.6280, je nižšia ako 2, preto aj metrika KLDIV úspešne validuje náš model vrámci datasetu.

#### 4.3.1.8 Coutrot 1 - NSS

Výsledná hodnota NSS je 0.6277. Hodnota NSS metriky odhaluje, že hodnoty v oblastiach fixácií korelujú s normalizovanými fixáciami používateľov ale na tomto type videí ma model ešte rezervu.

#### 4.3.1.9 Coutrot 2



**Obr. 4.8:** Vizualizácia všetkých testovaných metrík pre dataset Coutrot 2[7] pre jednotlivé videá

#### 4.3.1.10 Coutrot 2 - AUCROC

Výsledná priemerná hodnota pre AUC 0.6782.

#### 4.3.1.11 Coutrot 2 - KLDIV

Výsledná hodnota KLDIV je 1.8288 je nižšia ako 2, preto aj metrika KLDIV úspešne validuje náš model vrámci datasetu.

#### 4.3.1.12 Coutrot 2 - NSS

Výsledná hodnota NSS je 2.2313. Vysoká hodnota tejto metriky na tomto type videí je logickým dôvodom typu videa. V tomto datasete bola pozornosť podľa výstupných meraní užívateľov upriamená na iba málo oblastí v obraze naraz (v tomto prípade tvár alebo ruky účinkujúcich) a týmto oblastiam pridam vysokú hodnotu. Navrhovaná metóda postupuje podobne, vyčlení pohybujúce sa oblasti (pre takýto typ videa je pohybuje sa málo) a keďže netrovi majoritnú oblasť videa utlačí statické príznaky, čoho dôsledkom je vysoké hodnotenie týchto oblastí čo odpovedá vysokej hodnote porovnávajúcej sa v metrike.



#### **4.3.1.13 Zhrnutie hodnotenia**

Pri celkovom hodnotení bude pre nás smerodajná hlavne metriky AUCROC. Priemer pre všetky 3 testované datasety má hodnotu 0.6277 z čoho jasne vyplýva úspešná validácia korelácie s nameranými dátami. Metrika KLDIV dosahuje priemernú hodnotu pre všetky datasety 1.6438. Metrika KLDIV poukazuje na fakt že metóda označuje viac oblastí ako významných aj keď neboli namerané. Metrika NSS dosahuje priemernú hodnotu pre všetky datasety 0.5487. Táto metrika validuje, že aj hodnoty v oblastiach fixácií korelujú s dátami. V tejto metrike však záleží od typu videa viac ako u iných metrík.

#### **4.3.2 Porovnávanie s konkurenčnými modelmi pozornosti**

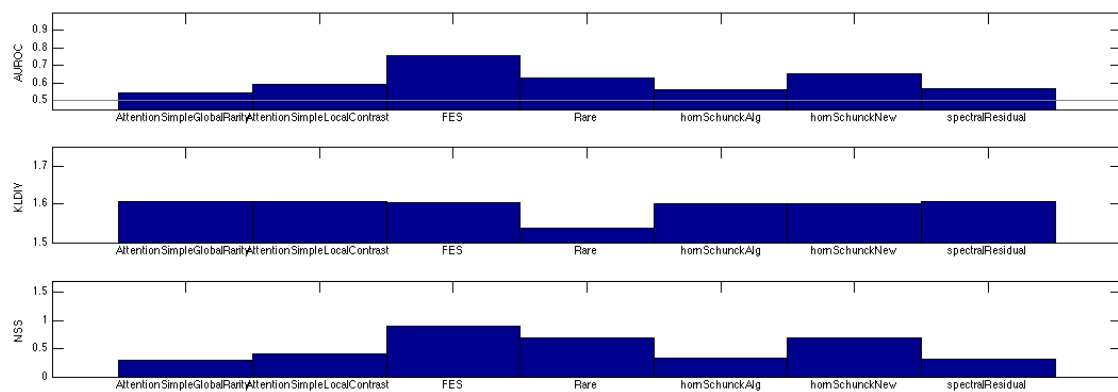
Porovnanie s konkurenciou nám poskytne ďalšie relevantné poznatky. Budeme sa konkrétne snažiť dokázať že efektívnosť nášho nového modelu je vyššia ako pôvodného horn-struck algoritmu[2], používaného na extrakciu dynamickej zložky príznakov. Zároveň sa budem snažiť o dôkaz, že algoritmus je efektívnejší aj ako samotná statická zložka, ktorá je vypočítavaná pomocou modelu Spektrálnych rezidual[14]. Pre tento účel som uskutočnil benchmark obsahujúci nasledovné algoritmy:

1. **AttentionSimpleGlobalRarity[17]**
2. **AttentionSimpleLocalContrast[17]**
3. **FES[19]**
4. **RARE[21]**
5. **Horn-struck[2]**
6. **Lucas-Kanade[4]**
7. **Spektrálne rezidua[14]**

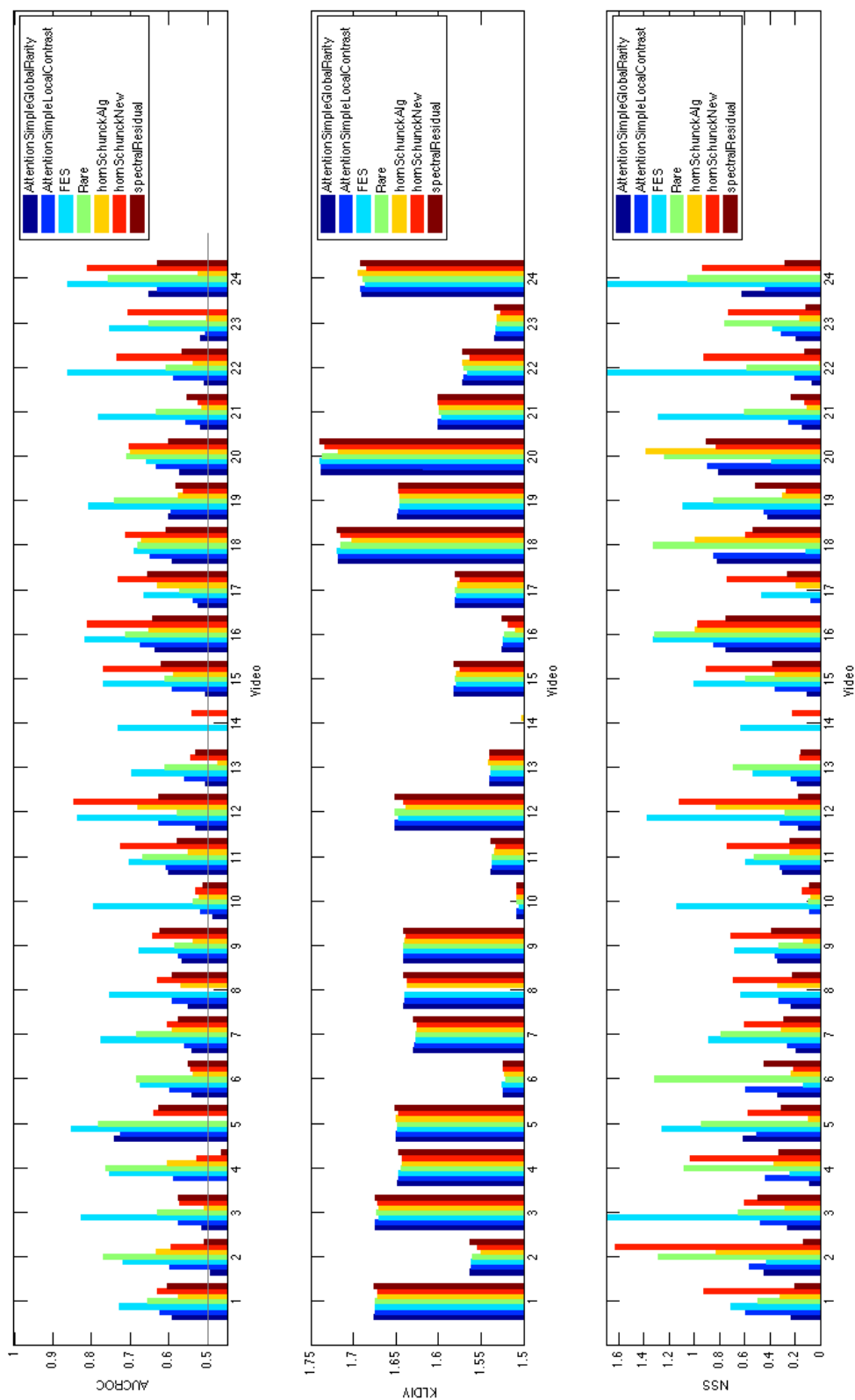
na všetkých datasetoch. Ku každému datasetu uvedieme 2 súhrnné štatistiky. Prvá vyjadruje porovnanie priemernej hodnoty každej z metrík pre každé video samostatne, druhá obsahuje priemerné hodnoty všetkých videí, pre všetky testované metódy.

##### **4.3.2.1 ASCMN**

V ASCMN datasete bolo testovanie uskutočnené na všetkých videách a všetkých vyššie uvedených modeloch.



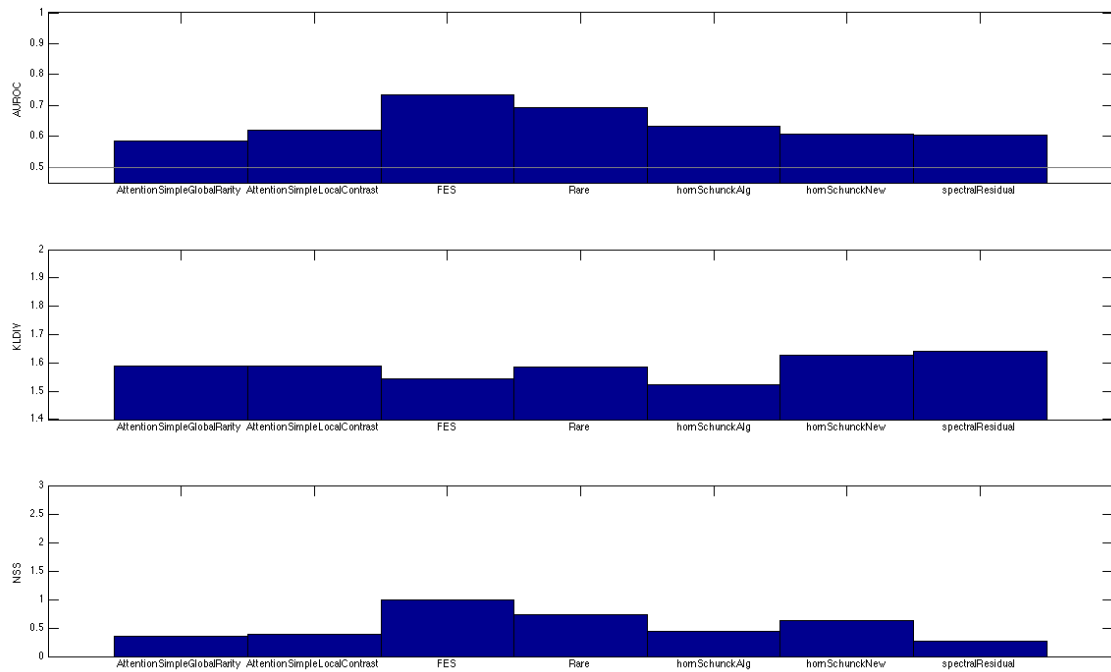
**Obr. 4.9:** Vizualizácia porovnania pre dataset ASCMN[22]



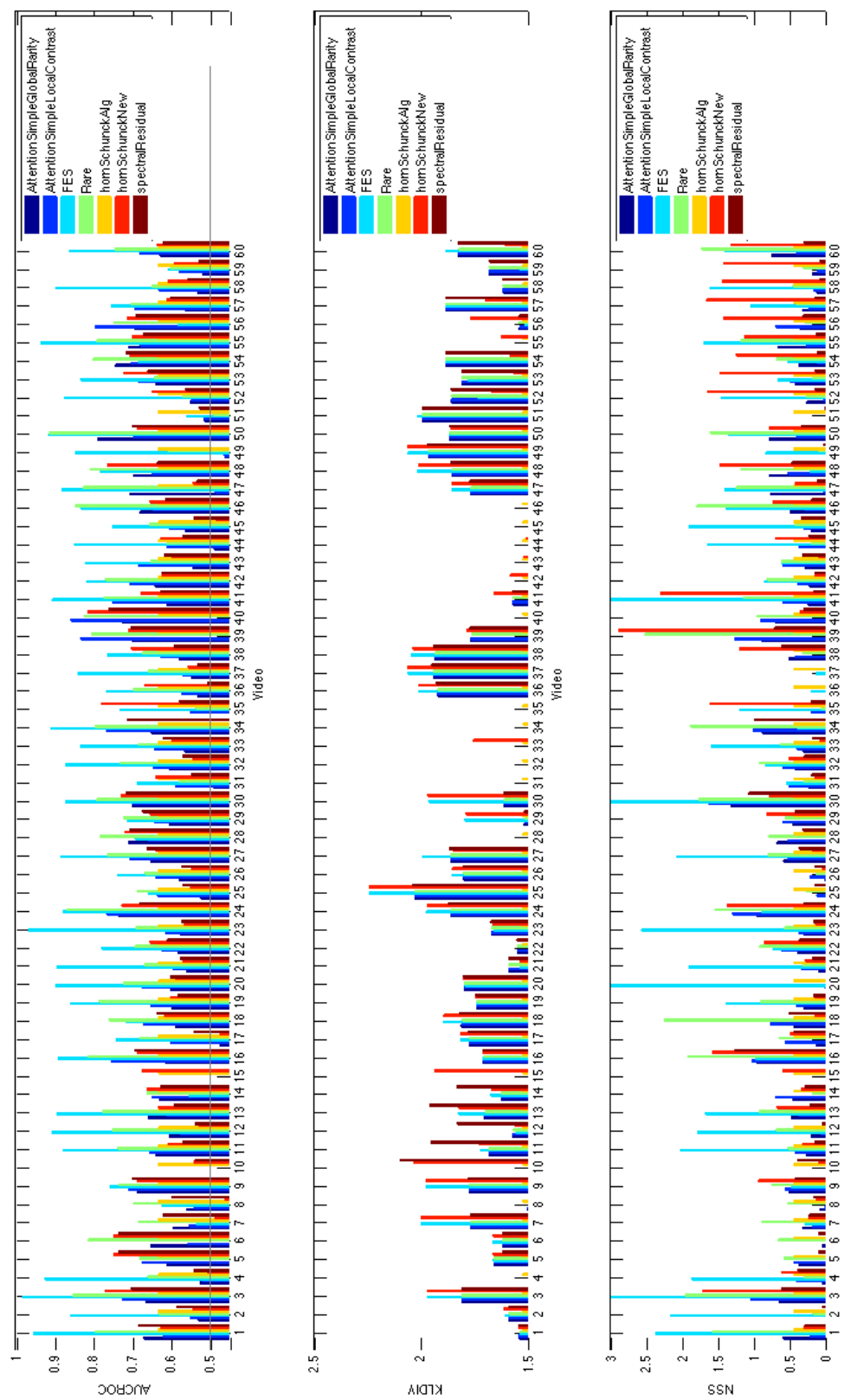
Obr. 4.10: Vizualizácia porovnania pre dataset ASCMN[22]

#### 4.3.2.2 Coutrot 1

V datasete Coutrot 1 boli z testovania vynaté videá číslo: 10, 15 z dôvodu chybového spracovanie v jednom alebo viacerých testovaných modelov.

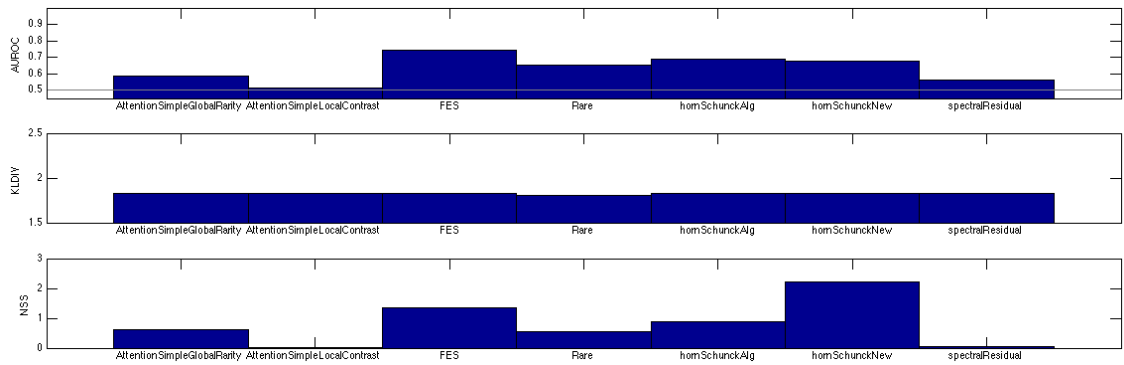


**Obr. 4.11:** Vizualizácia porovnania pre dataset Coutrot 1[9]



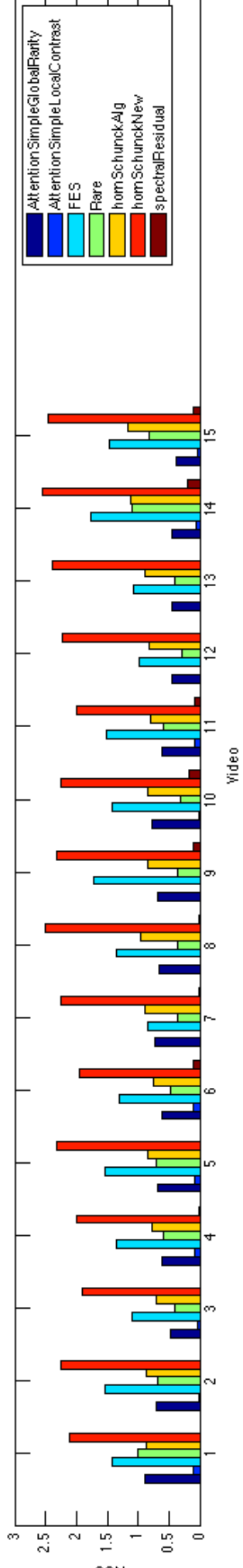
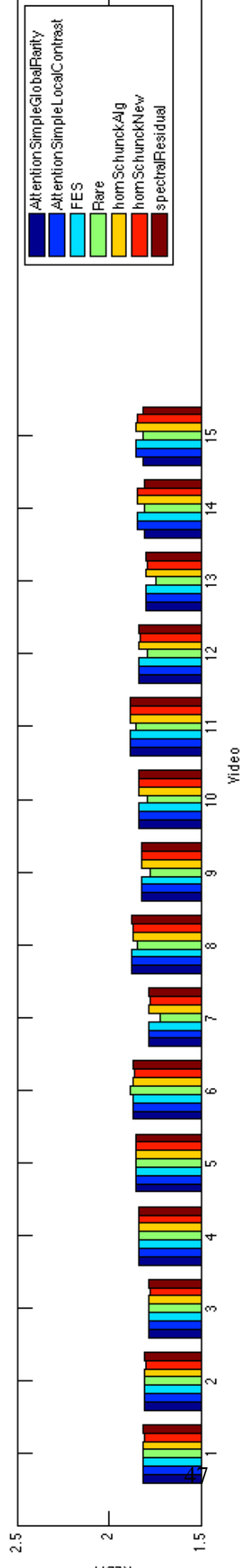
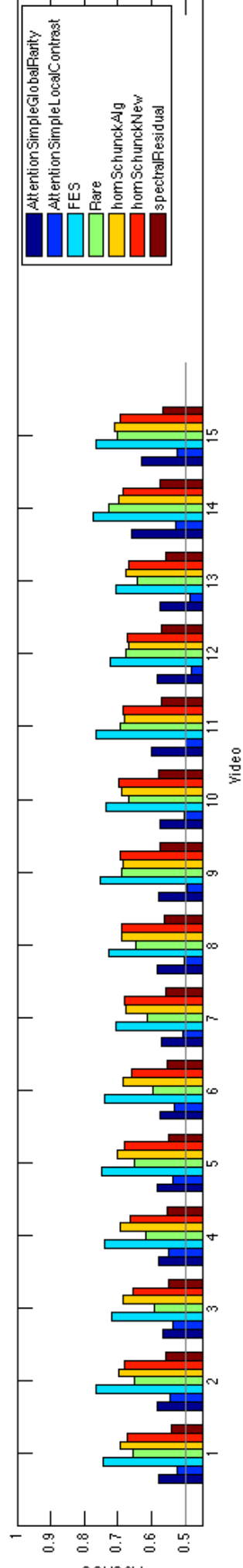
**Obr. 4.12:** Vizualizácia porovnania pre dataset Coutrot 1[9]

### 4.3.2.3 Coutrot 2



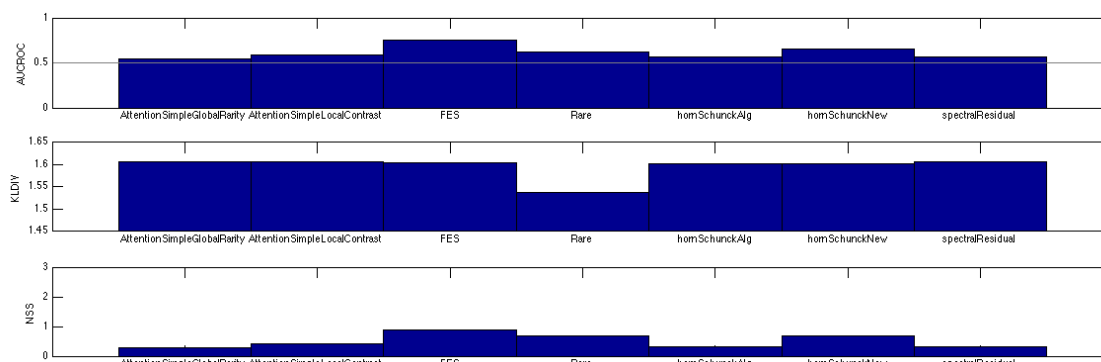
**Obr. 4.13:** Vizualizácia porovnania pre dataset Coutrot 2[7]

V datasete Coutrot 2 bolo testovanie uskutočnené na všetkých videách a všetkých vyššie uvedených modeloch.



#### 4.3.2.4 Zhrnutie benchmarku

Celkové hodnotenie benchmarku znázorním porovnaním priemerov všetkých datasetov spolu.



Obr. 4.15: Vizualizácia porovnania pre všetky datasety

#### 4.3.3 Zhrnutie validacie

Všetky metriky potvrdzujú tvrdenie, že navrhovaný model ma značnú koreláciu k skutočným dátam nameraných na reálnych užívateľoch. Zároveň validácia bola prevedená na typovo rozdielnych videách, keďže videá obsahujú pohybujúcu sa kameru aj statickú pozíciu kamery, konverzačné scény aj scény s prírodnými motívami. Zároveň na základe vypracovaného benchmarku môžeme tvrdiť, že nová metóda je efektívnejšia vo väčšine prípadov ako obe základné metódy použité na získanie dynamických aj statických príznakov a vo všetkých videách dosahuje lepšie výsledky ako aspoň jeden zo základných algoritmov.

### 4.4 Diskusia

Možnosť na zlepšenie algoritmu vidieť v celom benchmarku, kde model Rare[21] dosahoval výrazne lepšie výsledky aj napriek používaniu iba statických príznakov. Preto použitie algoritmu Rare[21] by viedlo k zlepšeniu aktuálnych výsledkov.

Ďalšia možnosť ako vylepšiť je v rýchlosti spracovania, ktorá nie je použiteľná na realtime spracovanie videa v štandardnej kvalite videa. Algoritmus je svojou časovou náročnosťou vhodný na spracovanie videí v nízkej obrazovej kvalite. Avšak pri vysokej obrazovej kvalite, algoritmus nevykazoval vyššiu efektivitu (otestované na datasete savam[13] ktorý poskytuje videá vo vysokej kvalite). Ale výpočet trval výrazne dlhšie ako v porovnaní s videom s nízkou obrazovou kvalitou.

Ďalšiu možnosť pre zlepšenie odhaľuje validácia datasetu Coutrot 2[7], kde pôvodný algoritmus horn-struck dosiahol hodnotenie porovnateľné s navrhovaným modelom. Takéto výsledky sú spôsobené výberom rozostupu frameov (podľa ktorých sa počíta dynamická zložka). Keďže rozpätie bolo zvolené na každú dvojicu framov pri týchto videách sa často



stávalo, že algoritmus detekoval iba minimálny pohyb. Čo bolo považované za šum a z toho dôvodu bola dynamická zložka potlačená alebo úplne eliminovaná (čo bolo v tomto prípade chybné). Riešením by bolo porovnávanie viacej framov a následná extrakcia pohybu všetkých dvojíc dokopy. Tento postup by už nebol považovaný za šum a dynamická zložka by nebola eliminovaná. Ďalšou možnosťou je použiť nejakú formu rozloženia videa na dynamicky sa meniace keyframey medzi ktorými sa bude výpočítavať dynamická zložka mapy, zatiaľ čo statická sa bude generovať počas každého framu.

## 5. Záver

Cieľom diplomovej práce bolo preskúmať možnosti extrakcie príznakov významných oblastí pre videá, pomocou príznakov ktoré nemožno extrahovať z čisto obrazovej informácie a navrhnúť metódu používajúcu takéto príznaky. Ďalším cieľom bolo vypracovať validáciu navrhovaného modelu pomocou štandardných metrík používaných pri hodnotení modelov na detekciu významných oblastí.

V práci som navrhol a adekvátne zvalidoval metódu, ktorá využíva kombináciu štandardných metód pre detekciu významných oblastí vo videu. Navyše bol vypracovaný benchmark, ktorý obsahuje 6 iných modelov/algoritmov za účelom porovnania a analýzy výsledkov navrhovaného modelu. Sekundárnym prínosom práce je ucelená aplikácia, ktorá má potenciál výrazne zjednodušiť prototypovanie, testovanie a validovanie podobných modelov pre budúcich študentov. Aplikácia je navrhnutá pre výrazné ušetrenie práce, vďaka jednoduchému pridávaniu modelov pozornosti a možnosti ich vizuálneho porovnania. Taktiež je jej pomocou možné validovať výsledky pomocou oficiálne publikovaných 3 datasetov a 3 štandardne používaných metrík. Validácia prebieha automaticky, a dokáže z výsledkov aj generovať výsledky vo forme grafov používaných aj v tejto práci. Aplikácia je dostupná v elektronickej prílohe a voľne dostupná na internete.

Ďalší možný rozvoj je možný popísaný a rozanalyzovaný v sekcii 4.4.

## Zoznam použitej literatúry

- [1] “A threshold selection method from gray-level histograms”, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, s. 62–66, Jan. 1979, ISSN: 0018-9472. DOI: 10.1109/TSMC.1979.4310076.
- [2] AL KANAWATHI, J. - MOKRI, S.S. - IBRAHIM, N. - HUSSAIN, A. - MUSTAFA, M.M. “Motion detection using horn schunck algorithm and implementation”, in *Electrical Engineering and Informatics, 2009. ICEEI '09. International Conference on*, vol. 01 : Aug. 2009. S. 83–87. DOI: 10.1109/ICEEI.2009.5254812.
- [3] AN, Kwang-Hwan - LEE, Minho - SHIN, Jang-Kyoo, “Saliency map model based on the edge images of natural scenes”, in *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, vol. 1 : 2002. S. 1023–1027. DOI: 10.1109/IJCNN.2002.1005616.
- [4] B.D. LUCAS, & Kanade. 1981. *An Iterative Image Registration Technique with an Application to Stereo Vision*. [online]. : 1981. [cit. 8.4.2013]. Dostupné na internete: [https://www.ri.cmu.edu/pub\\_files/pub3/lucas\\_bruce\\_d\\_1981\\_1/lucas\\_bruce\\_d\\_1981\\_1.pdf](https://www.ri.cmu.edu/pub_files/pub3/lucas_bruce_d_1981_1/lucas_bruce_d_1981_1.pdf).
- [5] BYLINSKII, Zoya - JUDD, Tilke - BORJI, Ali - ITTI, Laurent - DURAND, Frédo - OLIVA, Aude - TORRALBA, Antonio, *Mit saliency benchmark*.
- [6] COUTROT, A. - GUYADER, N. “An audiovisual attention model for natural conversation scenes”, in *Image Processing (ICIP), 2014 IEEE International Conference on* : Oct. 2014. S. 1100–1104. DOI: 10.1109/ICIP.2014.7025219.
- [7] —, “An efficient audiovisual saliency model to predict eye positions when looking at conversations”, in *Signal Processing Conference (EUSIPCO), 2015 23rd European* : Aug. 2015. S. 1531–1535. DOI: 10.1109/EUSIPCO.2015.7362640.
- [8] —, “Toward the introduction of auditory information in dynamic visual attention models”, in *Image Analysis for Multimedia Interactive Services (WIAMIS), 2013 14th International Workshop on* : Jul. 2013. S. 1–4. DOI: 10.1109/WIAMIS.2013.6616164.

- [9] —, “Toward the introduction of auditory information in dynamic visual attention models”, in *2013 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)* : Jul. 2013. S. 1–4. DOI: 10.1109/WIAMIS.2013.6616164.
- [10] COUTROT, Antoine - GUYADER, Nathalie, “How saliency, faces, and sound influence gaze in dynamic social scenes”, *Journal of Vision*, vol. 14, no. 8, p. 5, 2014. DOI: 10.1167/14.8.5. eprint: /data/Journals/JOV/933549/i1534-7362-14-8-5.pdf. Dostupné na internete: +%20http://dx.doi.org/10.1167/14.8.5.
- [11] COUTROT, Antoine - GUYADER, Nathalie - IONESCU, Gelu - CAPLIER, Alice, “Video viewing: do auditory salient events capture visual attention?”, *annals of telecommunications-Annales des télécommunications*, vol. 69, no. 1-2, s. 89–97, 2014.
- [12] DUNCAN, K. - SARKAR, S. “Saliency in images and video: a brief survey”, *Computer Vision, IET*, vol. 6, no. 6, s. 514–523, Nov. 2012, ISSN: 1751-9632. DOI: 10.1049/iet-cvi.2012.0032.
- [13] GITMAN, Y. - EROFEEV, M. - VATOLIN, D. - ANDREY, B. - ALEXEY, F. “Semiautomatic visual-attention modeling and its application to video compression”, in *Image Processing (ICIP), 2014 IEEE International Conference on* : Oct. 2014. S. 1105–1109. DOI: 10.1109/ICIP.2014.7025220.
- [14] HOU, Xiaodi - ZHANG, Liqing, “Saliency detection: a spectral residual approach”, in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on* : Jun. 2007. S. 1–8. DOI: 10.1109/CVPR.2007.383267.
- [15] ITTI, L. - KOCH, C. - NIEBUR, E. “A model of saliency-based visual attention for rapid scene analysis”, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 11, s. 1254–1259, Nov. 1998, ISSN: 0162-8828. DOI: 10.1109/34.730558.
- [16] LI, Jia - TIAN, Yonghong - HUANG, Tiejun - GAO, Wen, “A dataset and evaluation methodology for visual saliency in video”, in *Proceedings of the 2009 IEEE International Conference on Multimedia and Expo, ICME'09, New York, NY, USA* : IEEE Press, 2009. S. 442–445, ISBN: 978-1-4244-4290-4. Dostupné na internete: <http://dl.acm.org/citation.cfm?id=1698924.1699033>.
- [17] MANCAS, M. - MANCAS-THILLOU, C. - GOSSELIN, B. - MACQ, B. “A rarity-based visual attention map - application to texture description”, in *2006 International Conference on Image Processing* : Oct. 2006. S. 445–448. DOI: 10.1109/ICIP.2006.312489.

- [18] PROF. ROBERT FISHER, Prof. James Crowley. 2005. CAVIAR. [online]. : 2005. [cit. 8.4.2013]. Dostupné na internete: <http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>.
- [19] REZAZADEGAN TAVAKOLI, Hamed - RAHTU, Esa - HEIKKILÄ, Janne, “Image analysis: 17th scandinavian conference, scia 2011, ystad, sweden, may 2011. proceedings”, in, Heyden, A. -Kahl, F., Eds. Berlin, Heidelberg : Springer Berlin Heidelberg, 2011. Ch. Fast and Efficient Saliency Detection Using Sparse Sampling and Kernel Density Estimation, s. 666–675, ISBN: 978-3-642-21227-7. DOI: 10 . 1007 / 978 - 3 - 642 - 21227 - 7 \_ 62. Dostupné na internete: [http://dx.doi.org/10.1007/978-3-642-21227-7\\_62](http://dx.doi.org/10.1007/978-3-642-21227-7_62).
- [20] RICHE, N. - DUVINAGE, M. - MANCAS, M. - GOSSELIN, B. - DUTOIT, T. “Saliency and human fixations: state-of-the-art and study of comparison metrics”, in *Computer Vision (ICCV), 2013 IEEE International Conference on* : Dec. 2013. S. 1153–1160. DOI: 10.1109/ICCV.2013.147.
- [21] RICHE, N. - MANCAS, M. - GOSSELIN, B. - DUTOIT, T. “Rare: a new bottom-up saliency model”, in *Image Processing (ICIP), 2012 19th IEEE International Conference on* : Sep. 2012. S. 641–644. DOI: 10.1109/ICIP.2012.6466941.
- [22] RICHE, Nicolas - MANCAS, Matei - CULIBRK, Dubravko - CRNOJEVIC, Vladimir - GOSSELIN, Bernard - DUTOIT, Thierry, “Computer vision – accv 2012: 11th asian conference on computer vision, daejeon, korea, november 5-9, 2012, revised selected papers, part iii”, in, Lee, K. M. - Matsushita, Y. - Rehg, J. M. -Hu, Z., Eds. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013. Ch. Dynamic Saliency Models and Human Attention: A Comparative Study on Videos, s. 586–598, ISBN: 978-3-642-37431-9. DOI: 10 . 1007 / 978 - 3 - 642 - 37431 - 9 \_ 45. Dostupné na internete: [http://dx.doi.org/10.1007/978-3-642-37431-9\\_45](http://dx.doi.org/10.1007/978-3-642-37431-9_45).
- [23] SHARMA, P. - CHEIKH, F.A. - HARDEBERG, J.Y. “Face saliency in various human visual saliency models”, in *Image and Signal Processing and Analysis, 2009. ISPA 2009. Proceedings of 6th International Symposium on* : Sep. 2009. S. 327–332. DOI: 10.1109/ISPA.2009.5297732.
- [24] ŠIKUDOVÁ, E. - ČERNEKOVÁ, Z. - BENEŠOVÁ, W. - HALADOVÁ, Z. - KUČEROVÁ, J. 2014. *Počítačové videnie. Detekcia a rozpoznávanie objektov*, first : Wikina, Livornská 445, 109 00 Praha 10, 2014. .
- [25] ZHANG, Lingyun - TONG, Matthew H. - MARKS, Tim K. - SHAN, Honghao - COTTRELL, Garrison W. “Sun: a bayesian framework for saliency using natural statistics”, *Journal of Vision*, vol. 8, no. 7, p. 32, 2008. DOI: 10 . 1167 / 8 . 7 . 32.

eprint: /data/Journals/JOV/933536/jov-8-7-32.pdf. Dostupné na  
internete: +%20<http://dx.doi.org/10.1167/8.7.32>.

# Prílohy

CD obsahujúce:

- Elektronickú verziu
- Zdrojáky
- atď