

**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 svojmu vedúcemu práce za...

## **Abstrakt**

Tu je text slovenskej verzie abstraktu

**Kľúčové slová:** *slovo1, slovo2, slovo3, slovo4*

## **Abstract**

Text anglickej verzie abstraktu

**Keywords:** *word1, word2, word3, word4*

# Obsah

<b>1</b>	<b>Úvod</b>	<b>18</b>
<b>2</b>	<b>Prehľad literatúry</b>	<b>20</b>
2.1	Úvod do problematiky . . . . .	20
2.2	Metody pre statické obrázky . . . . .	20
2.2.1	Baseline Center . . . . .	20
2.2.2	Hrany . . . . .	20
2.2.3	Ittiho model . . . . .	21
2.2.4	Spektrálne rezidua . . . . .	21
2.2.5	Sun Model . . . . .	22
2.2.6	Rare Model . . . . .	22
2.3	Metody pre videá . . . . .	23
2.3.1	Zohľadnenie audio informácie . . . . .	23
2.3.2	Detekcia pohybu . . . . .	25
2.3.3	Lucas Kanade . . . . .	25
2.3.4	Horn-Schunck . . . . .	27
2.4	Metriky úspešnosti . . . . .	27
2.4.1	NSS . . . . .	28
2.4.2	AUC-Judd . . . . .	28
2.4.3	KL-Div . . . . .	28
2.5	Referenčné datasety . . . . .	29
2.5.1	RSD . . . . .	29
2.5.2	SAVAM . . . . .	30
2.5.3	ACoutrot DATASET . . . . .	30
2.6	Porovnanie štandardných Metód . . . . .	30
<b>3</b>	<b>Špecifikácia</b>	<b>31</b>
3.1	Platforma pre riešenie . . . . .	31
3.2	Očakávané výsledky . . . . .	31



3.3	Ideálne Prípady . . . . .	31
3.4	Problémové Prípady . . . . .	31
<b>4</b>	<b>Implementácia</b>	<b>32</b>
4.1	Návrh metódy . . . . .	32
4.1.1	Dynamické príznaky videa . . . . .	32
4.1.1.1	Rozdiel smerových vektorov v horizontálnom smere . . .	33
4.1.1.2	Rozdiel smerových vektorov v vertikálnom smere . . . .	33
4.1.1.3	Rozdiel vo vzdialenosti . . . . .	34
4.1.1.4	Spájanie regiónov . . . . .	34
4.1.1.5	Starnutie objektov na scéne . . . . .	35
4.1.2	Statické príznaky videa . . . . .	35
4.1.3	Výsledné spojenie príznakov . . . . .	35
4.1.4	Zdrojové kódy modelu . . . . .	36
4.1.5	Pipeline metódy . . . . .	37
4.2	Implementácia riešenia . . . . .	38
4.2.1	Aplikáciu na porovnávanie a automatickú validáciu . . . . .	38
4.2.1.1	Oddelenie logiky testovania a logiky samotného modelu .	38
4.2.1.2	Simultálne sledovanie videa z viacerých modelov . . . .	38
4.2.1.3	Automatická validácia modelu . . . . .	38
4.2.1.4	Vizualizácia výsledkov validácie . . . . .	39
4.2.2	Implementácia modulu . . . . .	39
4.3	Validácia výsledkov . . . . .	39
4.3.1	Analýza výsledkov . . . . .	40
4.3.1.1	ACCV 2012 - AUCROC . . . . .	40
4.3.1.2	ACCV 2012 - KLDIV . . . . .	40
4.3.1.3	ACCV 2012 - NSS . . . . .	40
4.3.2	Porovnávanie s konkurenčnými modelmi pozornosti . . . . .	41
4.3.2.1	ACCV 2012 - AUCROC . . . . .	41
4.3.2.2	ACCV 2012 - KLDIV . . . . .	41
4.3.2.3	ACCV 2012 - NSS . . . . .	41
4.4	Možnosti pre zlepšenie . . . . .	42
4.5	Diskusia . . . . .	42
<b>5</b>	<b>Záver</b>	<b>43</b>
	<b>Zoznam použitej literatúry</b>	<b>46</b>

**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*

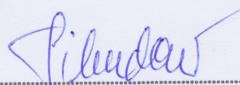
**Cieľ:** Metódy na detekciu významných oblastí vo videu:  
1. Naštudovanie  
2. Návrh zlepšenia  
3. Implementácia  
4. Porovnanie výsledkov

**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 svojmu vedúcemu práce za...

## **Abstrakt**

Tu je text slovenskej verzie abstraktu

**Kľúčové slová:** *slovo1, slovo2, slovo3, slovo4*

## **Abstract**

Text anglickej verzie abstraktu

**Keywords:** *word1, word2, word3, word4*



# Obsah

# 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[20]. 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 v výrazne rýchlejšie ako nefiltrované, ideálne v reálnom čase. Takéto oblasti sa nazývajú významné alebo charakteristické (v literatúre salient - prebrané v angličtine). Významné oblasti sú vyberané pomocou mnoha faktorov. Najznámejšími sú prechody vo farbe, intenzite alebo orientácií. Ľudský vizuálny systém taktiež využíva skúsenosti pri pozorovaní. Oblasť takto vyfiltrovaná nesú pre pozorovateľa viac potencionálnych informácií ako ostatné oblasti obrazu a preto sa stávajú salietnými.

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

Algoritmy na detekciu významných oblastí sa delia do 3 skupín podľa princípu akým spracovávajú dáta[10]

1. Zdola-nahor: Prístup je cielený na nezávislosť od používateľa. Zameriava sa 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 prechádzajúcich 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. Následne porovnanie nových metód s existujúcimi v rôznych štandardných oblastiach ako aj

v rýchlosti výpočtu.

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. Následne porovnanie s metódami uvedenými v prvej časti.

V závere....

## 2. Prehľad literatúry

### 2.1 Úvod do problematiky

Saliency a teda detekcia významných oblastí je využívaná v rôznych oblastiach. Počínajúc automatizáciou, modely významných oblastí (anglicky saliency modelov) sú ťažiskom pri segmentácii obrazu alebo detekcií špecifických objektov. Od saliency modelov sú taktiež závislé aj programy ovládajúce zabezpečovacie zariadenia. Tu salieny modely zužujú možnosti a proaktívne upozorňujú na podozrivé situácie. Až po reklamu, kde je vizuálna pozornosť kľúčovým parametrom, čo môže rozhodnúť o úspechu produktu, keď aký význam by mala reklama, kde si nevšimnete prezentovaný produkt alebo si všimnete iba jeho "menej" dokonalé časti.

### 2.2 Metody 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 nejefektívnejšie. 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.

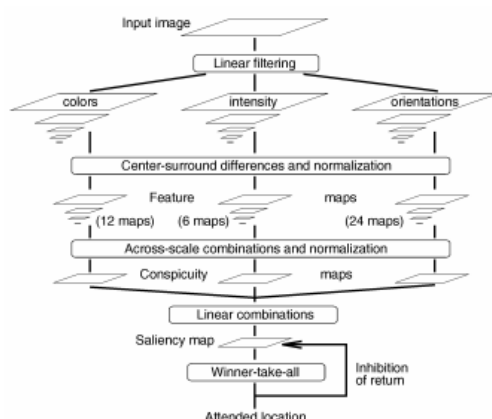
#### 2.2.2 Hrany

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

ľudského vyzuálneho 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 3 základné charakteristiky obrazu a to farbu, intenzitu, orientáciu.

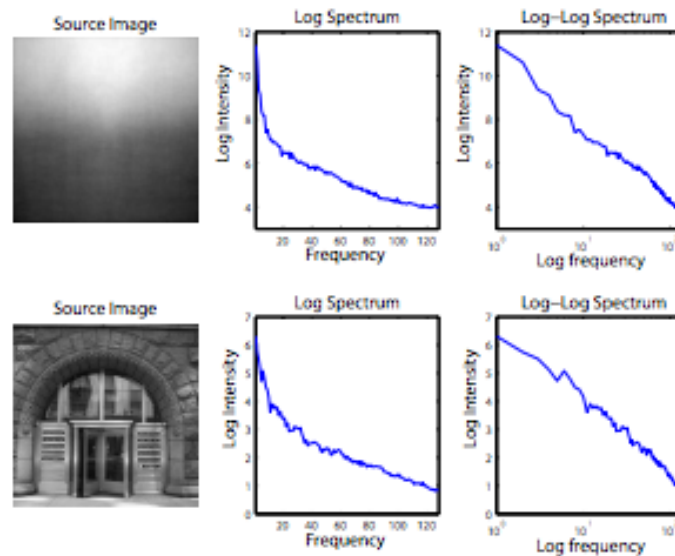


**Obr. 2.1:** Itti model general workflow.

Chrakteristika farby obsahuje 12 máp (šedotónové obrazy), pričom model používa farebný model RGB. Nazačiatku sa vypočíta intenzita podľa vzťahu  $I = (R + G + B)/3$ . Pomocou mapy I sa následne normalizujú všetky farebné kanály modelu RGB. Model extrahuje 4 farebné kanály červený (r), zelený (g), modrý (b), žltý (y) a pomocou Gausových pyramíd vytvorí 3 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 6 máp. Získaná je pomocou orientovaných gáborových filtrov s orientáciou  $0^\circ, 45^\circ, 90^\circ, 135^\circ$ . Dokopy 42 máp charakteristík je následne linárne skombinovaných do jednej saliency mapy[13].

### 2.2.4 Spektrálne rezidua

Medtoda 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

Amplitúdová zložka sa následne vyhladí, čím sa do popredia dostanú iba informácie, ktoré sa vymykajú z priemeru. Odčítaním od pôvodnej amplitúdoje zložky dostaneme iba časti obrazu, ktoré sú významné [12].

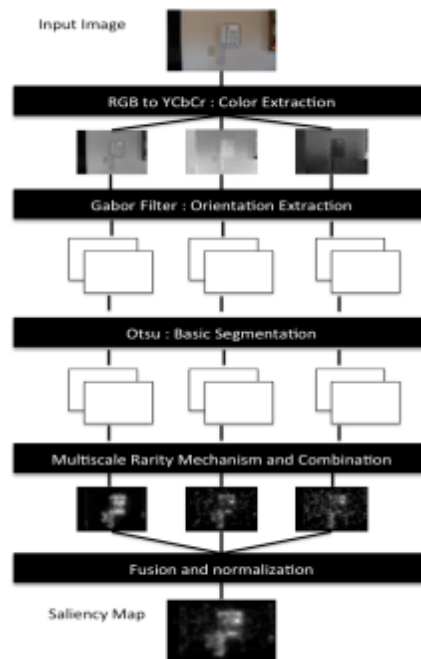
### 2.2.5 Sun Model

Sun model (Saliency Using Natural statistics) sa snaží simulovať potencionálne ciele sledovania ľudského vizuálneho systému. Model aktívne ohodnocuje 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[TODO referencia?]. Za výsledok hľadania potom udáva asymetrie v týchto štatistických štruktúrach[21].

### 2.2.6 Rare Model

Výrazná väčšina modelov pozornosti typu bottom-up funguje ustáleným postupom, kde sa z pôvodného obrazu extrahuje definovaná množina charakteristík paralelne a tie následne kombinujú alebo inak použijú na výpočet výslednej mapy pozornosti. Rare model navrhuje sekvenčnú architektúru, kde z pôvodného obrázku extrahuje nízko úrovňové príznaky. Následne na výsledkoch sériovo vykonáva extrakciu ďalších príznakov (v literatúre nazívané mid-level). Nakoniec ako posledný krok spojí a normalizuje výsledné charakteristiky do konečnej mapy významných oblastí. Rare model ako nízko úrovňové charakteristiky používa jas a colorimetrické rozdieli (ako farebný model používa YCbCr) a následne na mapách

rozložených žložiek farebného modelu detekuje orientáciu pomocou gáborových filtrov[17]. Po extrakcii všetkých charakteristík 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 nesequenčnej extrakcie príznakov z obrazu bol novým prístupom v oblasti modelov pozornosti.



Obr. 2.3: Rare model workflow

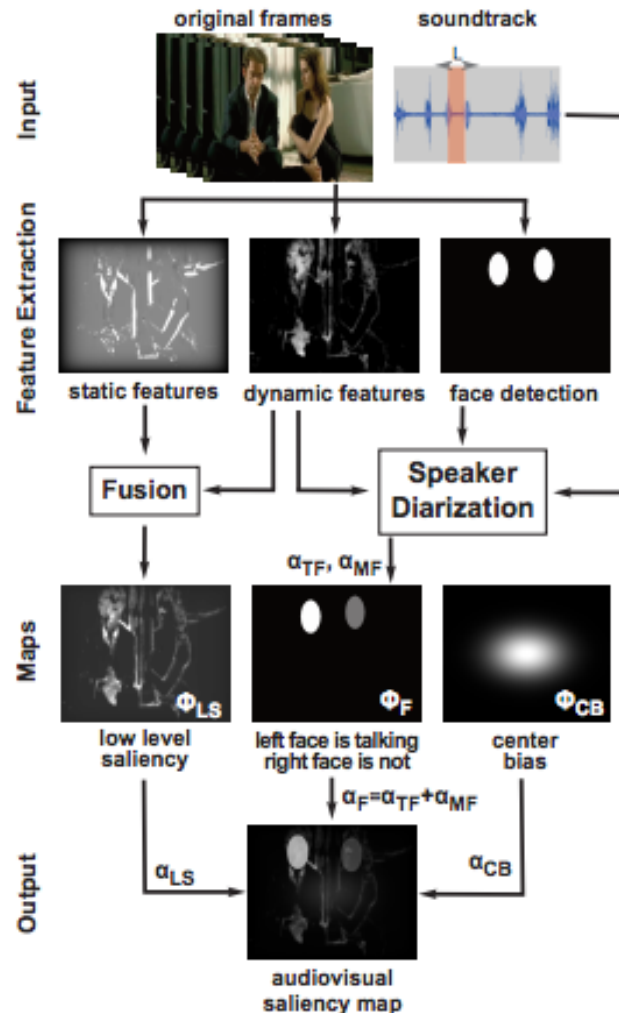
## 2.3 Metody pre videá

Video obsahuje rozsiahlejšie možnosti ako iba obrazová informácia, pribúdajú ďalšie rozmery ako je pohyb objektov na obraze alebo vplyv zvuku na ľudské vnímanie. Avšak oproti obrazu obsahuje potrebné spracovávať veďšie množstvo dát. Navyše vo väčšine algoritmov využívajúcich saliency modely je potrebné aby model dával výsledky v reálnom čase. Používané 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 [19]. Majoritná väčšina saliency modelov využíva iba obrazovú zložku ale zvuková stopa býva ponechaná stranou ale úplne zanedbaná. Použitie zvuku je známim trikom filmovej scény uz desatročia, kde režiséri posilňujú kontrolu nad diváckou pozornosťou práve pomocou zvukového doprovodu. Prvé štúdie sa zaoberali detekciou reči a tváre, kde je spojitost jednoznačná [8]. Neskoršie štúdie dokazujú korelácie aj na všeobecnejšej úrovni a pokusy o

extrakciu samotnej charakteristiky zo zvukovej stopy[9]. Tieto pokusy viedli aj k zostaveniu modelov zohľadňujúc zvukovú stopu ako samostatnú charakteristiku spolu s kombináciou s nízko-urovnovými príznakmi obrazu [6].



**Obr. 2.4:** Audiovisual model workflow.[6]

Model extrahuje video na sekvenciu obrazov (framy) a audio stopu v tvare grafu vlnovej dĺžky. Potom extrahuje 3 typy rôznych chrakteristík. Nízko-úrovňové príznaky založené na biologicky inširovaných saliency modeloch rozdelených na dynamickú časť a satickú časť. Statická časť sa zameriava na najasnejšie a najkontrastnejšie časti obrazu. Dynamická časť sa zameriava na relatívny pohyb objektov vhl'adom na pozadie (eliminácia pohybu kamery). Tieto 2 časti sa nakoniec spoja. Ďalšou chrakteritikou 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 druhej 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:

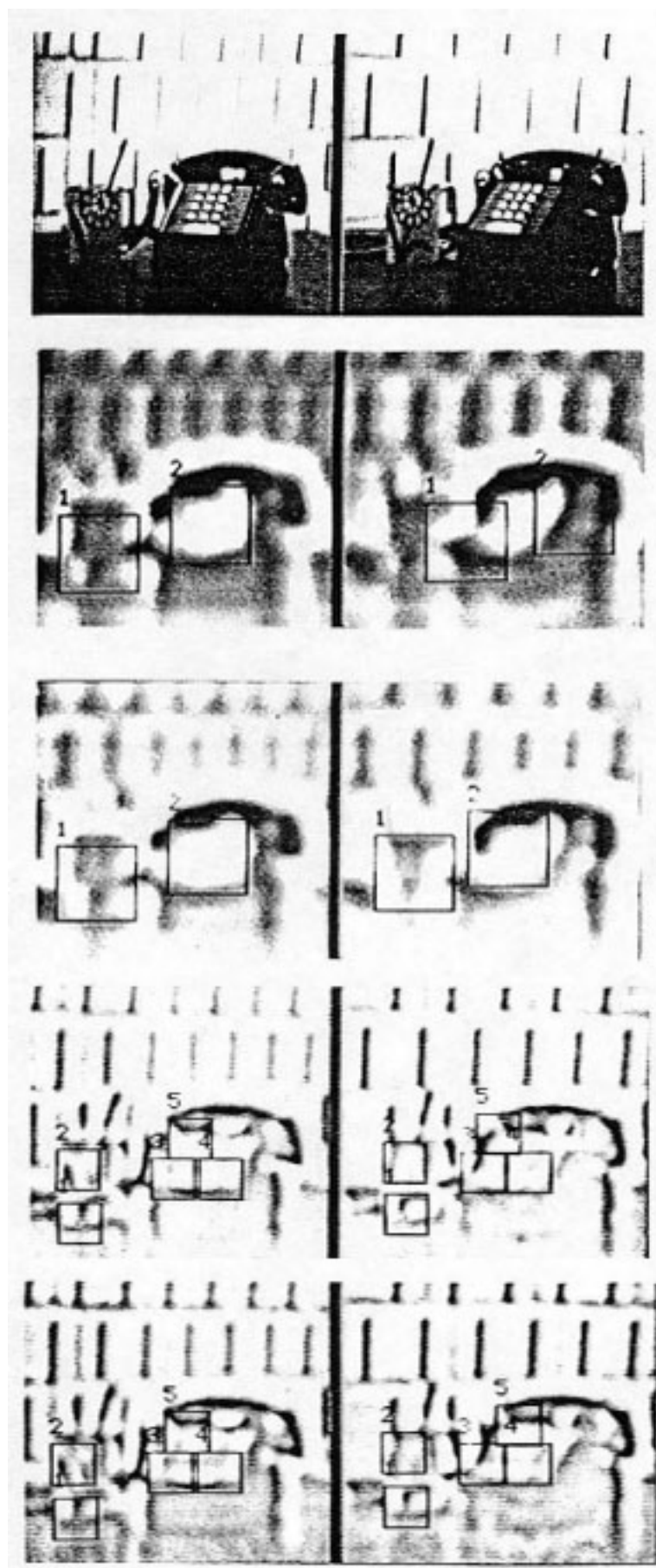
$$I(x + dx/dt, y + dy/dt, t + dt) = I(x, y, t) \quad (2.1)$$

Z čoho je ľahko odvodené, že:

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

### 2.3.3 Lucas Kanade

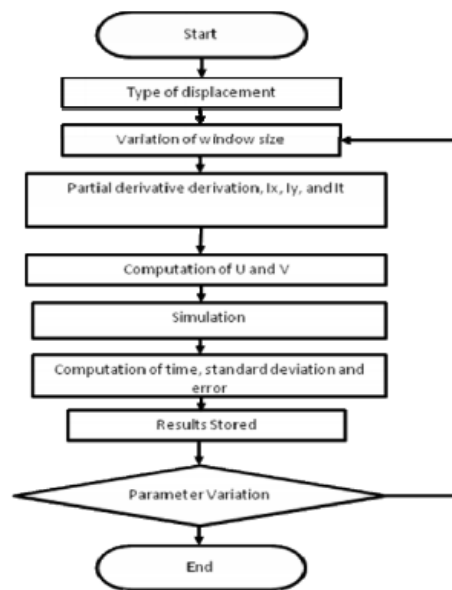
Algoritmus prvotne vznikol ako návrh pre časovú optimalizáciu problému výpočtu vektoru posunu medzi dvomi krivkami. Povodné intuitívne riešenie vyžadovalo  $O(M^2 * N^2)$  času pre výpočet daného vektoru ak M,N bolo rozlíšenie daného obrazového vzoru. Vtedy navrhovaná optimalizácia vyžadovala zadanie 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:** Vyzualizacie výsledkov algoritmu Lucas-Kanade vždy po 1 iterácii

### 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: Vyzualizácia 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[16].

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[16].

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

#### 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

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:

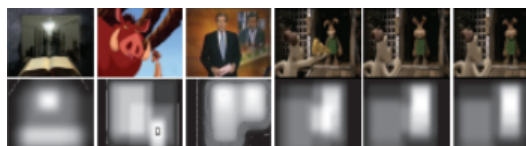
- **RSD**[14]
- **SAVAM**[11]
- **AUDITORY DATASET**[7]

### 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[15].
- **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[11].

### **2.5.3 ACoutrot DATASET**

Dataset žé videi pre vyhodnocovanie saliency modelov.

## **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 modeli pozornosti. V tejto sekcii ďalej uvedieme iba vyualizácie všetkých vyššie uvedených metód, pomocou obrázkov z testovaného videa (vybrané z vyššie uvedených datasetov) a im prislúchajúcich saliency máp vygenerovaných pomocou popísaných modelov alebo metód. Ďalej ich porovnanie pomocou výpočtových parametrov v tabulke.

## 3. Špecifikácia

### 3.1 Platforma pre riešenie

Ako platforma pre implementáciu budeme používať programovací jazyk Matlab.

### 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 rozmenu 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 modeli, 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 konkurenčnými 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

Najproblémovnejšími vstupmi očakávam videá s dynamickým pohybom kamery kombinovaným s pohybom objektov. 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.

## 4. Implementácia

### 4.1 Návrh metódy

Navrhovaná metóda zohľadňuje vlastnosti ktoré nie je možné získať iba zo statického obrazu, budeme ich nazívať dynamické príznaky videa. Avšak metóda stále zohľadňuje v pozorovanom videu aj aspekty statického obrazu, tieto budeme nazívať statické príznaky videa. Tieto príznaky sú vypočítavané seprátne 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 vektormy 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 kôli 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 akje počet väčší ako maximálna hodnota, zvýši konštantu o 10Tento 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. Pixeli s valídnuou honotou 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 regóny 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 zistujeme 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 sa 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

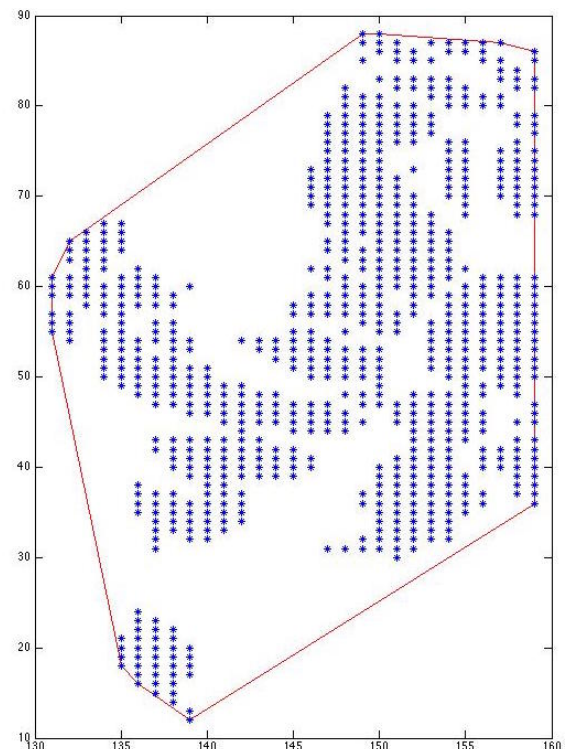
Chrakteristika sa vypočítava ako minimálna hodnota vzdialenosti medzi dvojicou regiónov. Hodnota je počítaná euklidovskou metódou.

```
forall the rohA ako každý extrém regiónu A do
| forall the rohB ako každý extrém regiónu b do
| | vzdialenost = 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 regionov, pre ktoré su všetky chrakteristky nižšie ako zadenfinovaná konštanta. Regióny spájame pomocou konvexného obalu zjednotenia bodov ležiacich v oboch regiónoch.



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

#### 4.1.1.5 Starnutie objektov na scéne

Do vypočítavnia 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 kdeže logicky označia celú scénu alebo jej väčšinu 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[12]. 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 s použijú v určitom pomere. Výpočet pomeru bude určovať pomer výskytu salientných pixelov v mape dynamických príznakov.

@TODO add latex som symbol

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

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,

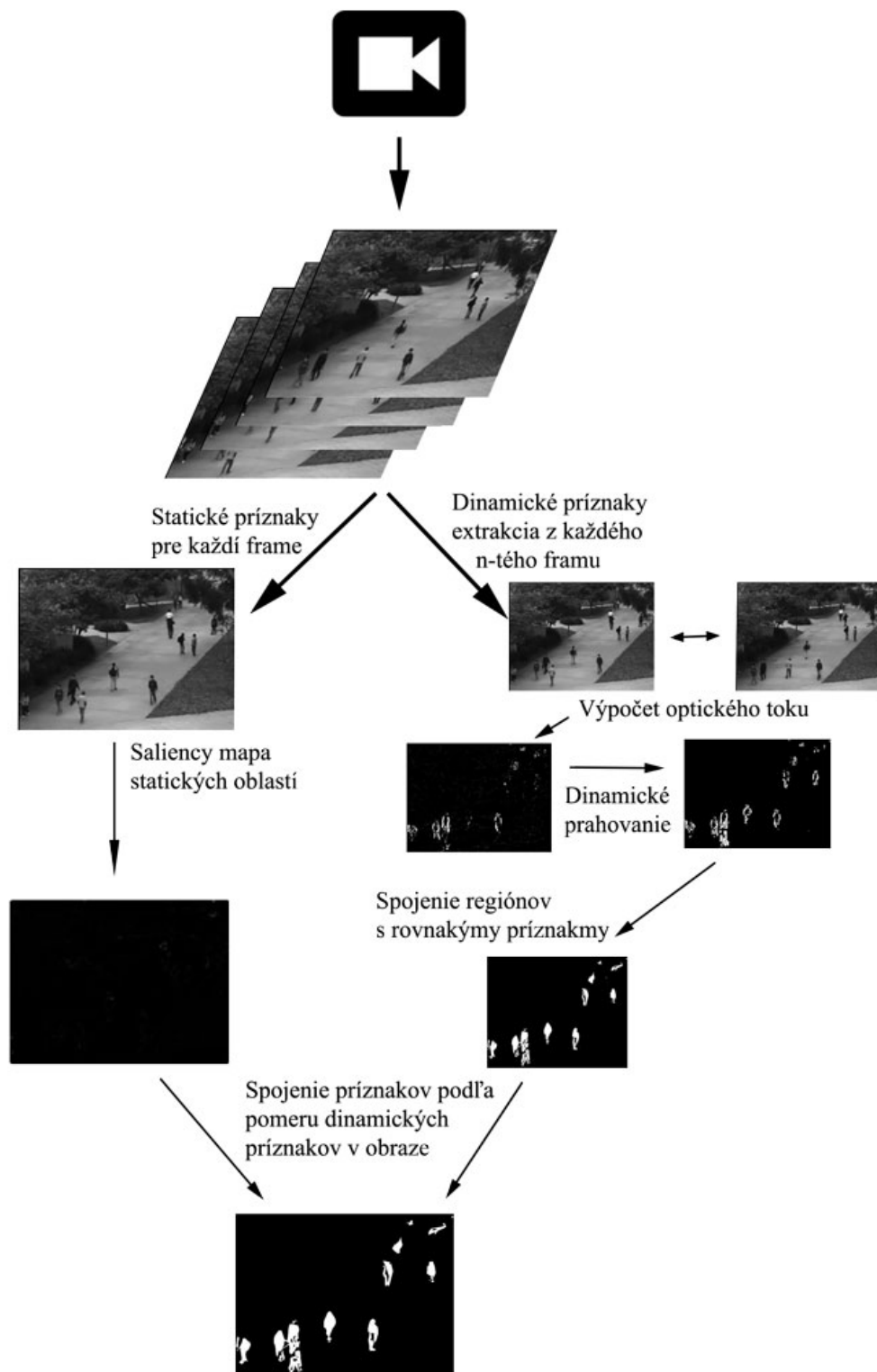
že kamera je v pohybe Horn-Schunck algoritmus označí ako pohybujúci sa väčšinou oblasť obrazu a v tom prípade je potrebné utlmiť dynamické príznaky obrazu a do popredia vystupujú statické.

#### **4.1.4 Zdrojové kódy modelu**

Zdrojový kód obsahuje jednu metódu ktorá priíma 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 používateľovy či použije model serializovane na každý frame videa, alebo zvolí vlastnú implementáciu keyframingu (napríklad kôli časovej náročnosti algoritmu). Algoritmus je schopný processovať farebné aj čiernobiele obrazové vstupy. V prílohe je možné nájsť 2 implementácie a to implementáciu ako 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, tátoimplementácia je určená na použitie mimo aplikácie na testovanie. Obe implementácie sú dostupné v prílohe na CD alebo voľne dostupné na internete.

### 4.1.5 Pipeline metódy

Grafický popis metódy obsahujúci príklad zostavenia mapy pozornosti.



Obr. 4.2: 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 aplikácií na testovanie je možné pripdávať ľubovoľné modeli pre ktoré je dostupná implmentácia v jazyku matlab. Pre iné jazyky je potrebné doprogramovať wrapper ktorý spustí daný jazyk a vypočíta mapu pozornosti. Ukážkový wrapper je súčasťou aplikácie. Pre pripadnie nového modelu je potrebné pridať wrapper do zložky "models", knižnice vyžadované modelmi je potrebné skopírovať do ľubovolnej 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: ACCV 2012[18], acoutrot( dataset nazov + referencia?), tieto datasety sú voľne dostupné a sú súčasťou aplikácie (zistiť či sú licencie s týmto ok). Dataset ACCV 2012[18] je poskytovaný autormi aj s testovacím algoritmom na výpočet vyššie uvedených metrík, do aplikácie na testovanie bol iba pozmenený pre načítanie ľubovoľného modelu a prispôbený na paralelný výpočet všetkých videí naraz. Pre dataset acoutrot aplikácia obsahuje upravenú verziu validačného algoritmu z datasetu ACCV 2012.

#### **4.2.1.4 Vizualizácia výsledkov validácie**

Pre analýzu výsledkov validácie dokáže aplikácia prehľadne vizualizovať všetky dáta získané testovaním.

### **4.2.2 Implementácia modulu**

Implementácia mového modelu pozornosti je jednoduchá. Pre integrovanie ľubovoľného modelu je možné použiť vzorovú implementáciu ktorá je dostupná v prílohách.

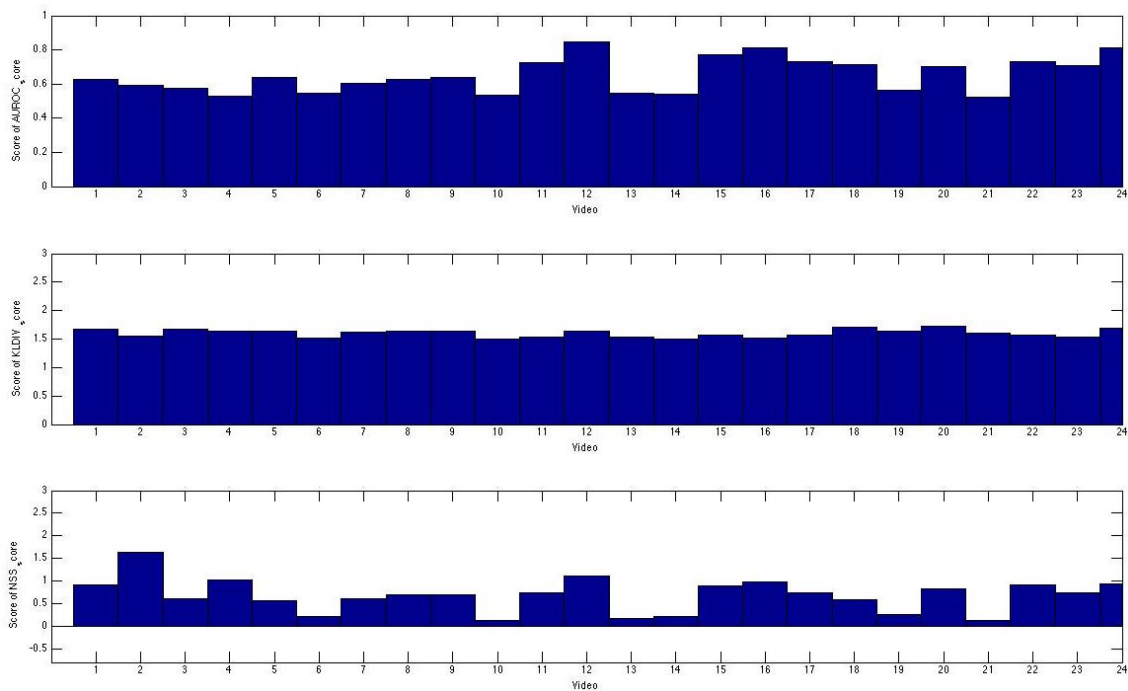
## **4.3 Validácia výsledkov**

Validácia vyššie spomínaného modelu prebiehala pomocou automatického testovania v aplikácií na testovanie. Validácia potvrdila, že detekcia pohybu a následné optimalizácie majú nenáhodný efekt vzhľadom na namerané hodnoty pomocou eyetrackingu viacerých používateľov na rôznych videách.

### 4.3.1 Analýza výsledkov

V tejto sekcií budem prezentovať výsledky všetkých testovaných datasetov. Výsledky budeme vizualizovať pomocou charakteristiky vzniknutej zo strednej hodnoty framov, pre jednotlivé videá datasetov. pre každú metriku samostatným grafom hodnôt.

Dataset: ACCV 2012[18]



**Obr. 4.3:** Vizualizácia všetkých testovaných metrík pre dataset ACCV 2012[18]

#### 4.3.1.1 ACCV 2012 - AUCROC

Podľa tejto metriky ...

#### 4.3.1.2 ACCV 2012 - KLDIV

Podľa tejto metriky ...

#### 4.3.1.3 ACCV 2012 - NSS

Podľa tejto metriky ...

Všetky metriky potvrdzuju tvrdenie, že navrhovaný model má značnú koreláciu k skutočným dátam namieraných na reálnych užívateľoch.

Dataset: coutrot1[18] @TODO images

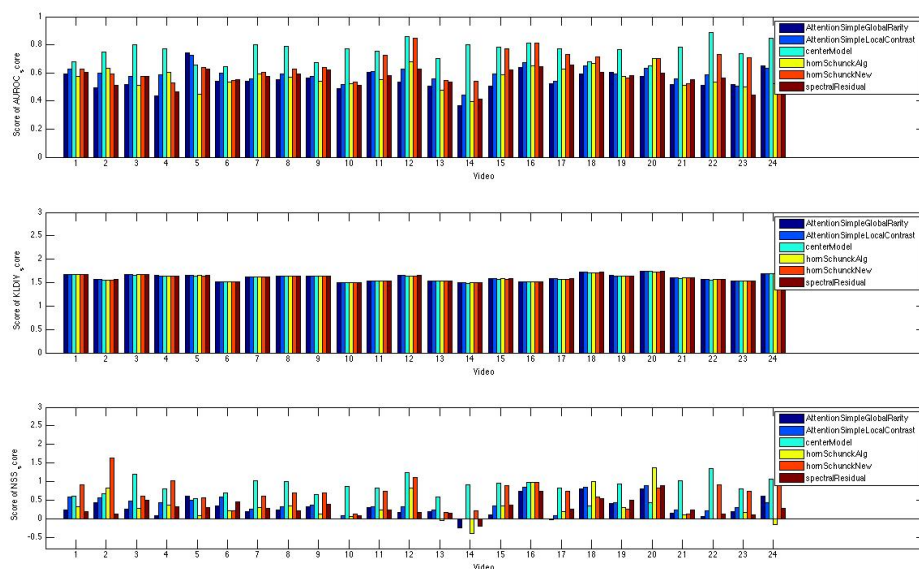
Dataset: coutrot2[18] @TODO



### 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 budeme snažiť o dôkaz, že algoritmus je efektívnejší aj ako samotná statická zložka, ktorá je vypočítavaná pomocou Spektrálnych rezidual[12].

Dataset: ACCV 2012[18]



Obr. 4.4: Vyzualizácia porovnania pre dataset ACCV 2012[18]

#### 4.3.2.1 ACCV 2012 - AUCROC

Podľa tejto metriky ...

#### 4.3.2.2 ACCV 2012 - KLDIV

Podľa tejto metriky ...

#### 4.3.2.3 ACCV 2012 - NSS

Podľa tejto metriky ...

Dataset: coutrot1[18] @TODO images

Dataset: coutrot2[18] @TODO

#### **4.3.3 Zhrnutie validacie**

#### **4.4 Možnosti pre zlepšenie**

#### **4.5 Diskusia**

## **5. Záver**

Cieľom diplomovej práce bolo...

V práci som.....

Ďalší možný rozvoj....

## 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] —, “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.
- [8] 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](http://dx.doi.org/10.1167/14.8.5).

- [9] 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.
- [10] 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.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] 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>.
- [15] 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/>.
- [16] 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.
- [17] 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.

- [18] 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).
- [19] 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.
- [20] 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. .
- [21] 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](http://data/Journals/JOV/933536/jov-8-7-32.pdf). Dostupné na internete: [+%20http://dx.doi.org/10.1167/8.7.32](http://dx.doi.org/10.1167/8.7.32).

# Prílohy

CD obsahujúce:

- Elektronickú verziu
- Zdrojáky
- atď