

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

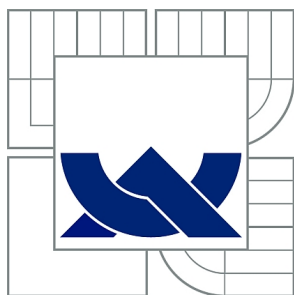
DE-IDENTIFIKACE OSOB NA ZÁKLADĚ PRIMÁRNÍCH
BIOMETRICKÝCH ZNAKŮ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

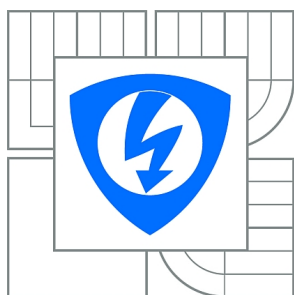
AUTOR PRÁCE
AUTHOR

BALÁZS ŠIDÓ

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNologiÍ**
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

DE-IDENTIFIKACE OSOB NA ZÁKLADĚ PRIMÁRNÍCH BIOMETRICKÝCH ZNAKŮ

PERSON DE-IDENTIFICATION UPON PRIMARY BIOMETRIC TRAITS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

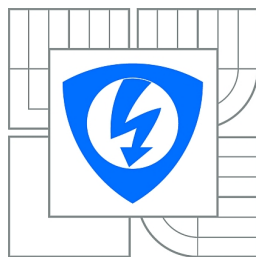
AUTOR PRÁCE
AUTHOR

BALÁZS ŠIDÓ

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ PŘINOSIL, Ph.D.

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Balázs Šidó

ID: 154885

Ročník: 3

Akademický rok: 2014/2015

NÁZEV TÉMATU:

De-identifikace osob na základě primárních biometrických znaků

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s možností de-identifikace osob na základě vybraných primárních biometrických znaků. Na základě získaných poznatků proveďte návrh vlastní metody pro de-identifikaci osob z multimediálních dat. Při návrhu metody zohledněte zachování co nejvyšší kvality původních dat a zároveň uvažte možnost reverzibility de-identifikačního procesu. Navrženou metodu implementujte ve vámi zvoleném programovacím jazyku a na reálných datech ověřte její vlastnosti.

DOPORUČENÁ LITERATURA:

[1] Li, S. Z., Jain, A. K.: Handbook of Face Recognition, Springer, 2nd ed. 2011, ISBN 978-0-85729-932-1.

[2] A. Senior.: Protecting Privacy in Video Surveillance, Privacy Protection in a Video Surveillance System, 35– 47, 2009.

Termín zadání: 9.2.2015

Termín odevzdání: 2.6.2015

Vedoucí práce: Ing. Jiří Přinosil, Ph.D.

Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato bakalářská práce se zabývá různými metodami pro ochranu soukromí osob pomocí de-identifikace na základě primárních biometrických znaků. První kapitola popisuje de-identifikace obličeje základními, primitivními a pokročilými metodami a dále obsahuje popis Aktivních modelů tvaru a vzhledu. Druhá kapitola je tvořena obecným přehledem zpracování řečových signálů a také metodami de-identifikace hlasu pomocí změny hladin energií v jednotlivých kmitočtových pásmech a pomocí změnou výšky hlasu. Další část tvoří popis realizovaného programu de-identifikace obličeje a návrh algoritmů pro reverzní de-identifikaci obličeje využitím parametru vzhledu Aktivních modelů vzhledu. Pak následuje návrh algoritmů pro de-identifikaci hlasu. V poslední části jsou uvedeny výsledky pokusů s de-identifikovanými obrazy. Programy byly realizovány v programovacím prostředí Matlab.

KLÍČOVÁ SLOVA

De-identifikace obličeje, De-identifikace hlasu, Aktivní model vzhledu, zachování kvality dat, ochrana soukromí

ABSTRACT

This bachelor's thesis deals with various methods protecting privacy by de-identification based on the primary biometric traits. The first chapter describes face de-identification with ad-hoc, naive and advanced methods. It also describes the Active Appearance Models and the Active Shape Models. The second chapter consists of a general overview of voice signal processing and of voice de-identification methods by changing the levels of energy in each frequency band and by pitch shift. The following chapters are describing the realized face de-identification program and an algorithm design for reversible face de-identification using the appearance parameters of the Active Appearance Model. This is followed by algorithm designs for voice de-identification. The results of the experiments done with the de-identified pictures are introduced in the last chapter. The realized program were written in Matlab.

KEYWORDS

Face De-identification, Voice De-identification, Active Appearance Model, data utility preservation, privacy protection

ŠIDÓ, Balázs *De-identifikace osob na základě primárních biometrických znaků*: bakalářská práce. BRNO: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 53 s. Vedoucí práce byl Ing. Jiří Přinosil, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „De-identifikace osob na základě primárních biometrických znaků“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

BRNO

.....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Jiřímu Přinosilovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

BRNO

.....

(podpis autora)

Výzkum popsany v této bakalářské práci byl realizovaný v laboratořích podpořených projektem Centrum senzorických, informačních a komunikačních systémů (SIX); registrační číslo CZ. 1.05/2.1.00/03.0072, operačního programu Výzkum a vývoj pro inovace.

OBSAH

Úvod	11
1 De-Identifikace Obličejů	12
1.1 Předzpracování obrazů	12
1.2 Základní metody de-identifikace obličejů	13
1.2.1 De-Identifikace pomocí odstranění obrazu	13
1.2.2 De-identifikace pomocí průměrování obrazu	14
1.2.3 Pixelace	14
1.2.4 Rozmazání	14
1.2.5 Primitivní metody de-identifikace	16
1.2.6 Nedostatky základních metod de-identifikace	18
1.3 Pokročilé metody de-identifikace obličejů	18
1.3.1 De-Identifikace pomocí k -Same	18
1.3.2 De-Identifikace pomocí k -Same-Select	19
1.3.3 Nedostatky k -Same	19
1.4 Aktivní modely vzhledu a tvaru	20
1.4.1 Aktivní modely tvaru	20
1.4.2 Aktivní modely vzhledu	21
2 De-Identifikace Řeči	22
2.1 Rychlá Fourierova Transformace	22
2.2 Melovské frekvenční keprstrální koeficienty	22
2.3 Dynamické borcení času	23
2.4 Ekvalizéry	24
2.5 De-identifikace pomocí změnou energií kmitočtových pásem	24
2.6 De-identifikace pomocí změny výšky hlasu	25
2.6.1 Změna délky trvání řečového signálu	25
2.6.2 Změna výšky zvuku	25
3 Realizace programu pro de-identifikaci obličejů	26
3.1 Program - Předzpracování	27
3.2 Program - De-identifikace obličejů	27
4 Návrh metody pro reverzní de-identifikaci obličejů	33
4.1 Vytvoření AAM	34
4.2 Reverzní „primitivní“ de-identifikace obličejů	36

4.3	Reverzní „pokročilá” de-identifikace obličeje	37
5	Návrh metody pro de-identifikaci hlasu	40
5.1	Napodobení hlasu jiného mluvčího	41
5.2	Změna výšky hlasu	45
6	Experimenty	47
7	Závěr	49
	Literatura	50
	Seznam symbolů, veličin a zkratk	52
A	Přílohy	53
A.1	Obsah přiloženého CD	53

SEZNAM OBRÁZKŮ

1.1	Předzpracování obrazu	12
1.2	Metoda Zatemnění	13
1.3	Metoda Průměrování	14
1.4	Pixelace	15
1.5	Rozmazání	15
1.6	Primitivní metody de-identifikace	17
1.7	k -Same	19
1.8	Modely tvarů [8]	21
1.9	Modely vzhledů	21
2.1	Ekvalizéry [17]	24
3.1	Obrazy z databáze <i>GUFD</i> [1]	26
3.2	Vypočtené souřadnice	27
4.1	Blokové schéma reverzní metody de-identifikace	33
4.2	Nastavení velikosti masky a vytvoření <i>ASM</i>	34
4.3	Rozdíly mezi původním obrazem a modelem <i>AAM</i>	35
4.4	Aplikace funkcí <i>exp</i> a <i>sin</i> na parametr vzhledu c	36
4.5	Aplikace Hammingova okénka na parametr c	37
4.6	Skutečný obsah buněk parametru c	37
4.7	Výsledné de-identifikované obrazy obličeje	39
5.1	Blokové schéma de-identifikace hlasu	40
5.2	Ukázka nejkratší cesty mezi dvě matice	41
5.3	Matice obsahující <i>FBE</i> původní a nové délky	42
5.4	Vygenerovaná melovská banka 13 filtrů	43
5.5	Znázornění filtrů v kaskádě [17]	43
5.6	Kmitočtové spektra původních dvou signálů a výstupního signálu	44
5.7	Kmitočtové spektra po změně výšky hlasu	46

SEZNAM KÓDŮ

3.1	Podprogram Bar mask	28
3.2	Část podprogramu Average	29
3.3	Podprogram Pixelace	29
3.4	Podprogram Rozmazání	30
3.5	Podprogram Negative	30
3.6	Podprogram Random Noise	31
3.7	Podprogram Treshold	31

ÚVOD

Nedávné pokroky v technologiích nám umožňují uchovávání a sdílení velkého množství dat. Na základě našich nahraných dat, nás může útočník identifikovat, a tím narušit naše soukromí. Identifikovat osobu je možné pomocí znalostí primárních biometrických znaků subjektu, tj. podle obličeje, hlasu, otisku prstu. Obrazy obličejů většiny lidí je možné nalézt na sociálních sítích. Existují ale případy, kdy identita osob viditelných na obrazech není podstatná, např.: Google Streetview. Zde je nutné de-identifikovat obraz. De-identifikace znamená odstranění informací z dat, podle kterých by se dalo identifikovat subjekt, před sdílením dat.

Množství soukromých dat, ke kterým se můžeme dostat, pomocí našeho hlasu narůstají. Transformace hlasu nám umožňuje změnit a tím napodobit hlas jiný. Pomocí transformace umí útočník změnit svůj hlas na hlas osoby, od které chce neprávem získat informace. Transformace hlasu je vhodná také pro de-identifikaci, protože podle transformovaného hlasu neumíme zpětně určit totožnost původního mluvčího.

Cílem této bakalářské práce je seznámit čtenáře s možnostmi de-identifikace obličeje a hlasu. V první kapitole této práce jsou popsány metody de-identifikace obličeje, kde jsou rozebrány základní, primitivní metody, metoda k -Same a její novější verze, které zanechávají vyšší kvalitu obličejů. Dále je zde uveden stručný popis Aktivních modelů vzhledu a tvaru. Druhá kapitola je tvořena úvodem do problematiky zpracování řečových signálů a jsou zde popsány dvě metody de-identifikace hlasu.

Další část se zabývá realizací a návrhem algoritmů na základě principů popsanych v první části. V třetí kapitole je popsán realizovaný de-identifikační program, který obsahuje všechny základní, primitivní a pokročilé metody de-identifikace popsány v první kapitole kromě metody k -Same-Select. Dále je uveden návrh metod de-identifikace obličeje pomocí Aktivních modelů vzhledu. V další kapitole následuje návrh metod de-identifikace hlasu pomocí změn energií v kmitočtových pásmech řečového signálu a pomocí změn výšek hlasu. V poslední kapitole jsou uvedeny výsledky pokusu s obrazy de-identifikovanými pomocí realizovaných programů.

1 DE-IDENTIFIKACE OBLIČEJE

De-identifikace obrazu obličeje může poskytnout jistou ochranu soukromí, ale samotná de-identifikace nezajišťuje absolutní ochranu. V obraze můžou zůstat detaily, které umožňují re-identifikaci subjektu, proces může být reverzibilní, dále existují další re-identifikační metody, které umožňují identifikaci. De-identifikace se využívá například při maskování identity lidí během rozhovorů v televizi. [7, 9]

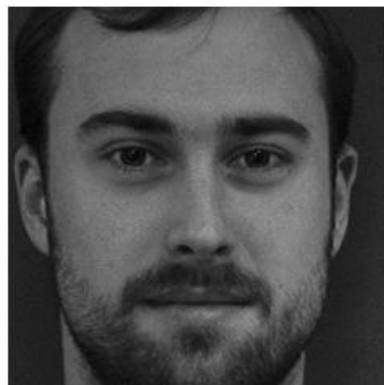
1.1 Předzpracování obrazů

Obraz obsahující obličej je nejprve převeden do formátu, kde každý obraz je reprezentován jako matice $\mathbf{P}$ s velikostí $m_1 \times m_2$. Každá buňka v matici $\mathbf{P}$ ukládá hodnotu od 0 do 255, což je hodnota znamenající intenzitu světlosti pixelu. Cílem předzpracování je upravit obraz tak, aby pozice očí a jejich středová vzdálenost byla zhruba na stejném místě v každém obraze, čímž se obrazy stávají srovnatelné. Každý obraz je převeden, normalizován, otáčen a oříznut podle potřeby. Tímto způsobem jsou vytvořeny obrazy obličejů.

Obraz obličeje je reprezentován maticí $\mathbf{O}$ s velikostí $n_1 \times n_2$. Každá buňka v matici $\mathbf{O}$ ukládá hodnotu od 0 do 255, což je hodnota znamenající intenzitu světlosti pixelů. Obraz obličeje 1.1b je vytvořen z obrazu 1.1a a je zde vidět, že obraz obličeje je oblast původního obrazu obsahující tvář subjektu. [9]



(a) Originální obraz



(b) Obraz obličeje

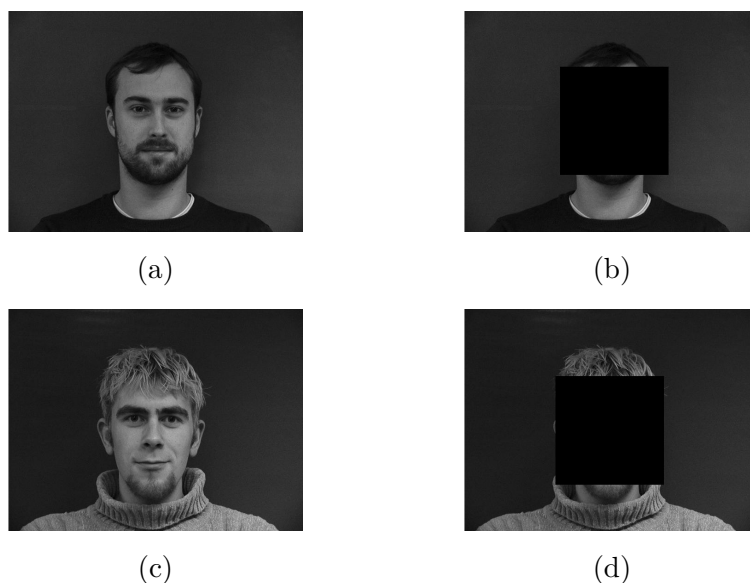
Obr. 1.1: Předzpracování obrazu

1.2 Základní metody de-identifikace obličeje

Tradiční tisk a vysílací media pro de-identifikaci obličeje používají metody potlačení dat nebo jednoduché filtrace obrazů. Filtrace obrazů využívá jednoduché zatemňovací metody jako rozmazání (vyhlazení obrazu např. s konvolucí obrazu Gaussovým filtrem) nebo pixelace (podvzorkování obrazu). Jednoduché filtrace ale nejsou dostatečně bezpečné, protože existuje program, který dokáže obraz obličeje zpětně identifikovat. Zatímco tyto algoritmy můžou být aplikovány na všechny obrazy, neposkytují dostatek soukromí. Z tohoto důvodu není garantováno, že soukromí lidí zobrazených na obrázku je opravdu ochráněno. [5]

1.2.1 De-Identifikace pomocí odstranění obrazu

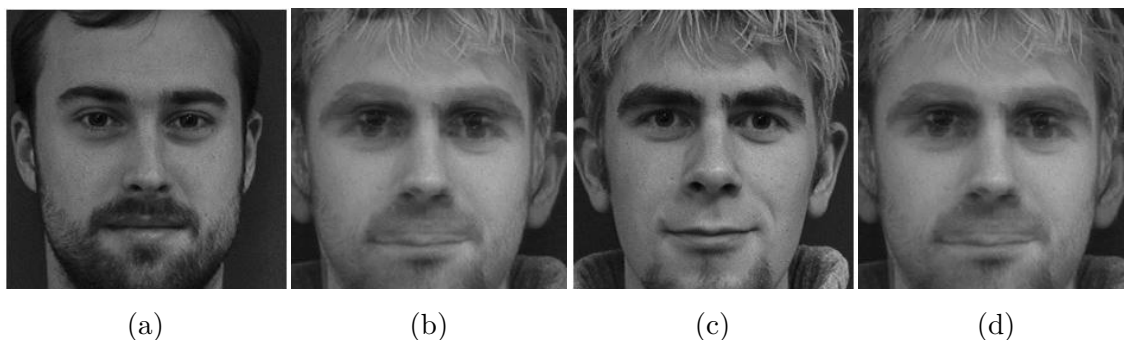
Metoda Zatemnění vrátí obraz ve kterém má každý pixel hodnotu 0. V černobílých obrazech, pixely s nulovou hodnotou se zobrazují černě. Obrazy 1.2b a 1.2d jsou obrazy po de-identifikaci pomocí metody Zatemnění. Obrazy 1.2a a 1.2c jsou původními obrazy. Ani počítač ani člověk neumí poznat, jestli subjekt kopie obrazu obličeje vytvořen z 1.2b nebo 1.2d, byl původně obraz 1.2a nebo 1.2c, protože obrazy obličeje jsou identické. Správné rozpoznání obličeje je limitováno na hádání s pravděpodobností $1/H$, kde H je počet obrazů de-identifikovaných pomocí této metody, tzn. v tomto případě $H = 2$. Zatemnění je tedy bezpochybně efektivní ochrana osobních údajů, kde nejsou zachovány žádné detaily obličeje. Přesto Zatemnění není nejlepší volbou de-identifikace v některých případech. V některých situacích je požadována efektivní de-identifikace, která přitom zachová detaily obličeje. [9]



Obr. 1.2: Metoda Zatemnění

1.2.2 De-identifikace pomocí průměrování obrazu

Metoda Průměrování vrátí obraz, ve kterém každý pixel v de-identifikovaném obrazu je suma pixelů na stejných pozicích původního obrazu a jiného obrazu nebo více obrazů vydělen celkovým počtem obrazů. Obrazy obličeje 1.3a a 1.3c jsou obrazy obličeje. Obrazy obličeje 1.3b a 1.3d jsou obrazy po de-identifikaci pomocí metody Průměrování. Počítač ani člověk neumí poznat, jestli subjekt kopie obrazu 1.3b nebo 1.3d byl původně obraz 1.3a nebo 1.3c, protože 1.3b a 1.3d jsou identické. Jak po metodě Zatemnění tak po metodě Průměrování je správné rozpoznání obličeje limitováno na odhad s pravděpodobností $1/H$. To znamená, že Zatemnění a Průměrování jsou stejně efektivní, ale z pohledu zachování detailů v obraze je metoda Průměrování mnohem lepší než metoda Zatemnění. [9]



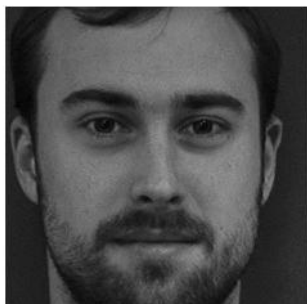
Obr. 1.3: Metoda Průměrování

1.2.3 Pixelace

Proces Pixelace zredukuje informace obsahující obraz podvzorkováním. Pro daný faktor pixelace p , dílčí bloky obrazu s rozměrem $p \times p$ jsou extrahované a nahrazené s průměrnou hodnotou pixelů z dílčích bloků. Se stoupající se hodnotou p je z obrazu odstraněno více informací, viz obrazy 1.4. [4]

1.2.4 Rozmazání

Rozmazání obrazu znamená, že každý pixel v obrazu je nahrazen váženým průměrem pixelů v jeho okolí. Používaná volba Rozmazání je Gaussova funkce, která váží pixely blízko svého okolí. Pro dvě dimenze, souřadnice x a y , Gaussův operátor rozmazání je vypočten pomocí rovnice 3.2. Směrodatná odchylka σ určuje velikost okolí. Rozmazaný obraz je pak vypočten, jako konvoluce původního obrazu s Gaussovým operátorem rozmazání. [4] Se stoupající se hodnotou σ je výsledný obraz víc rozmazán, viz obrazy 1.5.



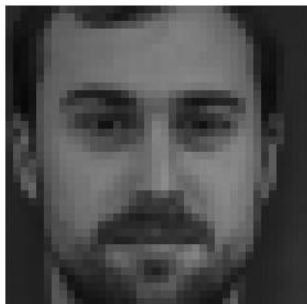
(a) Původní obraz



(b) $p = 2$



(c) $p = 4$



(d) $p = 6$

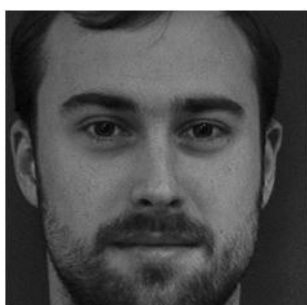


(e) $p = 10$

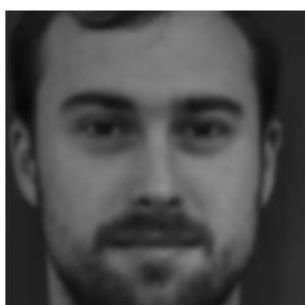


(f) $p = 15$

Obr. 1.4: Pixelace



(a) Původní obraz



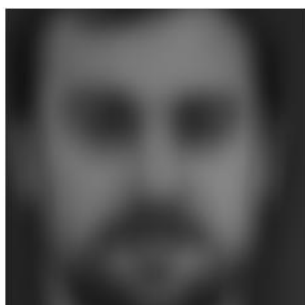
(b) $\sigma = 2$



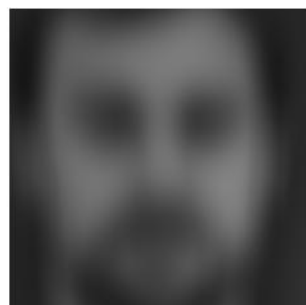
(c) $\sigma = 4$



(d) $\sigma = 6$



(e) $\sigma = 10$



(f) $\sigma = 15$

Obr. 1.5: Rozmazání

1.2.5 Primitivní metody de-identifikace

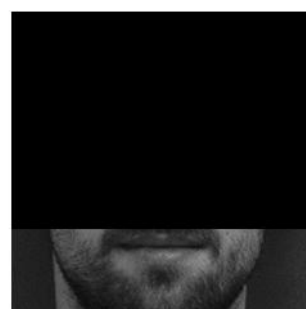
- **Bar mask:** Pokrytí očí jednobarevným pásem, viz obraz 1.6a.
- **T mask:** Pokrytí očí a nosu jednobarevným pásem ve tvaru T, viz obraz 1.6b.
- **Mouth only:** Pokrytí celého obličeje, nekryté části jsou jen ústa a brada, viz obraz 1.6c.
- **Negative:** Pro obrazy ve stupních šedi je změněna hodnota x každého pixelu na hodnotu $255-x$. Pro černobílý obraz platí, že každý černý pixel (0) se stává bílým (255) a naopak, viz obraz 1.6d.
- **Random Noise:** Hodnota náhodně vybraného pixelu je nahrazena náhodnou hodnotou. V obrazech ve stupních šedi je sada náhodně vybraných pixelů nahrazena náhodně vygenerovanou hodnotou mezi 0 a 255, viz obraz 1.6e.
- **Ordinal:** Hodnota každého pixelu je přepočtena na jednu z pěti hodnot, viz obraz 1.6f.
- **Threshold:** Hodnoty pixelů jsou nahrazeny černou (0) nebo na bílou (255) barvou v závislosti na hodnotě t která je v rozmezích od 0 do 255, viz obrazy 1.6g, 1.6i a 1.6h.
- **Mouth only Černobílý:** Metoda kombinující metody Threshold a Mouth only. Nejprve je aplikována na obraz metoda Threshold, kde $t=128$, pak se pokryje celý obličej, nekryté části jsou jenom ústa a brada, viz obraz 1.6j.
- **Negative Černobílý:** Metoda kombinující metody Negative a Threshold, viz obraz 1.6k.
- **Random Noise Černobílý:** V obrazech ve stupních šedi náhodně vygenerovaná hodnota 0 nebo 255, nahradí sadu náhodně vybraných pixelů, viz obraz 1.6l. [4]



(a) Barmask



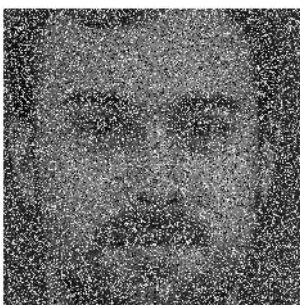
(b) T mask



(c) Mouth Only



(d) Negative



(e) Random Noise



(f) Ordinal



(g) Threshold $t=50$



(h) Threshold $t=100$



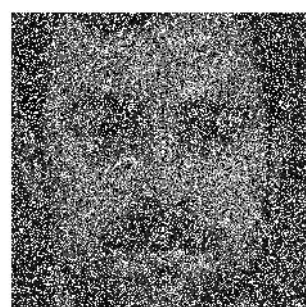
(i) Threshold $t=128$



(j) Mouth Only ČB



(k) Negative ČB



(l) Random Noise ČB

Obr. 1.6: Primitivní metody de-identifikace

1.2.6 Nedostatky základních metod de-identifikace

Zatímco základní de-identifikační metody jako Rozmazání a Pixelace prokázaly, že úspěšně dokážou zamezit rozpoznání člověkem, jsou zranitelné proti poměrně jednoduché technice rozluštění. Velmi efektivní přístup pro rozluštění obrazu základních de-identifikačních algoritmů byl nabídnut Newtonem a spol. [7] jménem papouškové rozpoznání. Prvním krokem manuální papouškové rozpoznávací metody je aplikace stejné zkreslení pro všechny obrazy ve vstupní sadě obrazů podle toho, jaké je zkreslení de-identifikovaného obrazu. Namísto toho, aby byl porovnán de-identifikovaný obraz s původními obrazy, je tento obraz porovnán de-identifikovanými obrazy. V důsledku se míra rozpoznání výrazně zlepší a ochrana osobních údajů poskytnuta základními algoritmy de-identifikace se sníží.

V případě manuální papouškové rozpoznání je míra de-identifikace předem známa a obrazy jsou de-identifikovány ve stejné míře. Avšak tato informace může být extrahována z de-identifikovaných obrázků pro automatický papouškové rozpoznání. V případě Pixelace, je určena velikost bloků se stejnou intenzitou pixelů v obrázku. V případě rozmazání se analyzuje frekvenční spektrum de-identifikovaných obrázků. [5, 7]

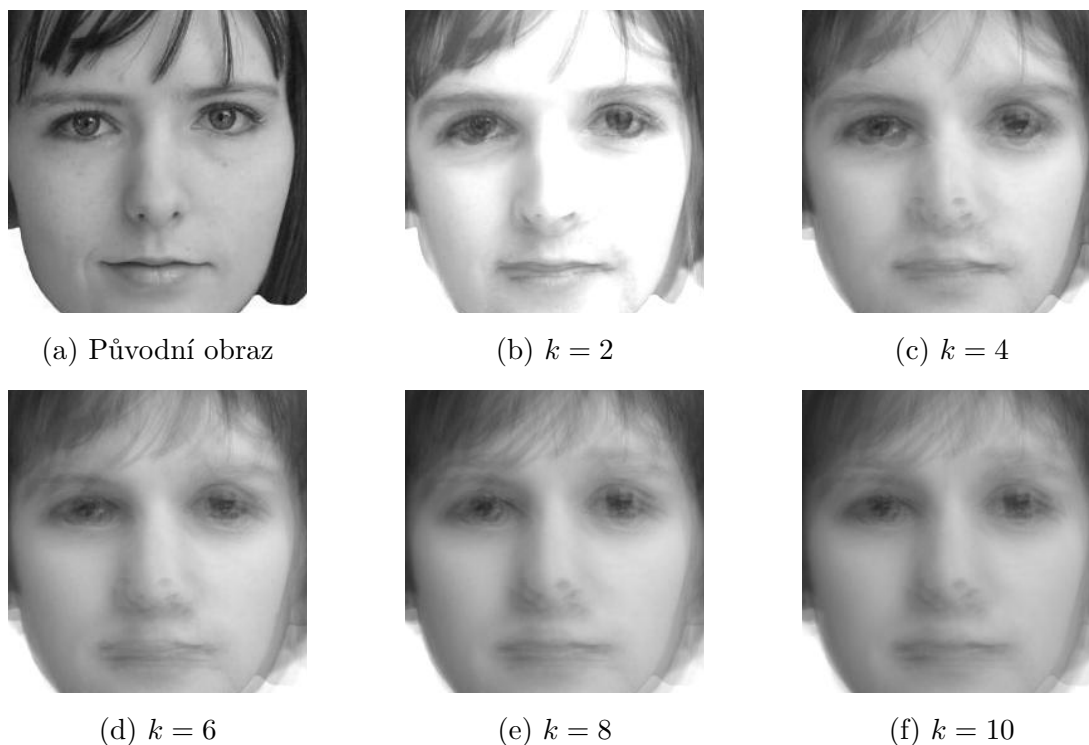
1.3 Pokročilé metody de-identifikace obličeje

Pokročilé metody de-identifikace obličeje jsou založeny na k -anonymitě, která je definována tak, že poskytuje ochranu soukromí tím, že každý záznam ve výsledné sadě obrazů se vztahuje alespoň ke k jednotlivcům. k -anonymita zajišťuje, aby pravděpodobnost úspěšné rozpoznání byla pod $1/k$. [9]

1.3.1 De-Identifikace pomocí k -Same

Rozdělení sady obličejů, kde každý subjekt je reprezentován jen jednou, do menších sad, do tzv. „skupin“ obsahující alespoň k obličejů, poskytuje ochranu soukromí, protože je vygenerován jeden souhrnný obraz z každé skupiny na místo originálních obrazů. Založení každého souhrnného obličeje z homogenních původních obrazů minimalizuje ztrátu informace. Obtížnějším částem metody je nalezení skupin s k obrazy, ve kterých jsou homogenní obličeje. Pro určení homogenity anebo blízkosti se používá měření vzdálenosti. Vzdálenost se nejčastěji určí podle eukleidovské vzdálenosti. De-identifikované obrazy obličeje musí být konstruovány takovým způsobem, aby se minimalizovala ztráta informace. Existují způsoby například nalezení ekvidistantního bodu-obličeje, nebo bod s nejmenší čtvercovou vzdáleností. Nalezení průměrného obličeje reprezentující danou skupinu je taky jedna možnost. Takový

obraz je nazýván centroidem. Centroid skupiny je obličej, který je vypočten průměrováním hodnot pixelů ve stejné pozici v každém obraze skupiny, viz obrázky 1.7. Metoda k -Same poskytuje k -anonymitu, s co nejmenší ztrátou informace, přičemž úspěšné rozpoznání obličeje je pod pravděpodobností $1/k$. [9]



Obr. 1.7: k -Same

1.3.2 De-Identifikace pomocí k -Same-Select

Jeden z nedostatků metody k -Same spočívá v tom, že obrazy jsou seskupené pouze na základě eukleidovské vzdálenosti, tzn. neumí rozpoznat výraz obličeje a pohlaví. Algoritmus k -Same-Select rozdělí vstupní sadu obrazů do vzájemně exkluzivních podskupin podle pohlaví, výrazu obličeje, a pak na tyto skupiny aplikuje algoritmus k -Same. Výsledný algoritmus poskytuje k -anonymitu, přičemž lépe zachovává užitečnost a kvalitu dat. [4]

1.3.3 Nedostatky k -Same

k -Same předpokládá, že každý subjekt se vyskytuje v sadě obličejů jenom jednou, případ, který v praxi není příliš častý. Protože k -Same používá nejbližších sousedů

daného obrazu při de-identifikaci, přítomnost vícenásobných počtů ze stejného obrazu ve vstupní sadě vede k nižší úrovni ochrany soukromí. Protože v mnoha případech de-identifikace se pracuje s individuálními obrazy anebo se sekvencemi obrazů, k -Same v takových případech není aplikovatelná. [5]

1.4 Aktivní modely vzhledu a tvaru

Cílem metod „interpretace syntézou“ je napodobit obrazy, vytvořením syntetických obrazů co nejvíce se podobajících. Aktivní modely tvaru *ASM* (*Active Shape Model*) spolu s Aktivními modely vzhledu *AAM* (*Active Appearance Model*) jsou schopné generovat napodobeniny určitých objektů (nejčastěji obličejů) pomocí parametrů, v nichž jsou obsaženy určité informace o tvarech a vzhledech. Aplikace této metody a přizpůsobení parametrů k obrazu, spočívá v minimalizaci rozdílu mezi vstupním obrazem a modelem. Parametry jsou pak použity pro další zpracování. Například pro rekonstrukci původního obličeje, nebo jsou tyto parametry podány programu pro rozpoznání obličeje.

Aktivní modely jsou výsledkem kombinací modelů tvaru s modely textury, kde textura je intenzita světlosti nebo barva jednotlivých pixelů. K vytvoření jednoho modelu, je potřeba v každém obrazu vstupní sady označit významné body. K vytvoření modelu obličeje se používají obrazy obličeje, kde jsou označeny hlavní rysy tváře. Aktivním modelem je možné napodobit obraz ve dvou krocích: Nejprve je model tvaru přizpůsoben ke krajním částem obličeje v obraze, pak modelem vzhledu je rekonstruována textura. [8]

1.4.1 Aktivní modely tvaru

Tvar Aktivního modelu je definován souřadnicemi významných částí obličeje, které společně vytvářejí masku obličeje. Matematicky lze tvar vyjádřit následovně:

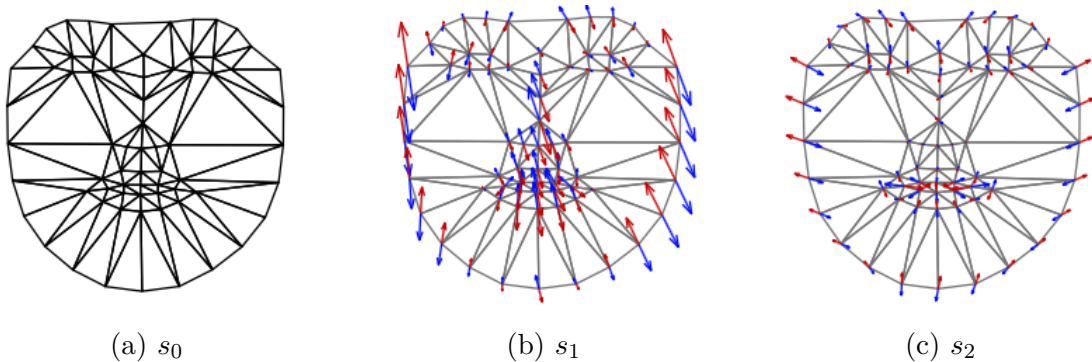
$$s_0 = (x_1, y_1, x_2, y_2, \dots, x_n, y_n), \quad (1.1)$$

kde x udává souřadnici pixelu v horizontální rovině, y udává souřadnici ve vertikální rovině. *ASM* umožňuje lineární variaci tvaru, což znamená, že výsledný tvar s může být vyjádřen jako základní tvar s_0 plus lineární kombinace několika tvarových vektorů s_i :

$$s = s_0 + \sum_{i=1}^n v_i s_i, \quad (1.2)$$

kde p_i je parametr tvaru.

ASM je obvykle vypočten z ručně označených vstupních obrazů. Modely tvarů se normalizují posunutím nebo změnou velikosti, aby každá souřadnice všech masek odpovídala poloze souřadnic základní masky. [8]



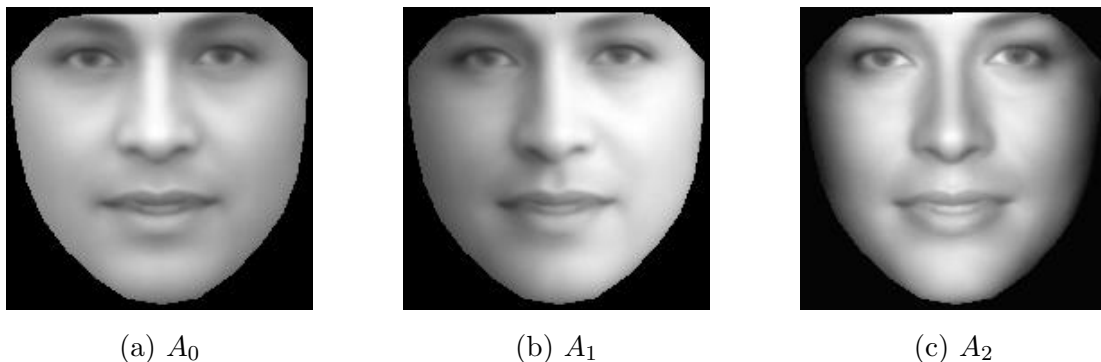
Obr. 1.8: Modely tvarů [8]

1.4.2 Aktivní modely vzhledu

Vzhled Aktivního modelu je definován v oblasti základní masky s_0 . AAM obsahuje pixely, které leží uvnitř s_0 . AAM je obraz $A(x)$ definován pixely x , pro které platí $x \in s_0$. AAM umožňuje lineární variaci vzhledu, což znamená, že výsledný tvar $A(x)$ může být vyjádřen jako základní vzhled $A_0(x)$ plus lineární kombinace několika vzhledových obrazů A_i :

$$A(x) = A_0(x) + \sum_{i=1}^n c_i A_i(x), \quad (1.3)$$

kde c_i je parametr vzhledu.



Obr. 1.9: Modely vzhledů

Každý AAM je normalizován deformací každého obličeje s ručně označenou maskou tak, aby se výsledný AAM co nejvíce podobal základnímu tvaru vzhledu. [8]

2 DE-IDENTIFIKACE ŘEČI

Řeč lze reprezentovat jako elektrický v čase proměnlivý analogový signál. Při konverzi analogového signálu na digitální tj. vzorkování, se signál vyjádří s čísly, které reprezentují úroveň signálu v daném okamžiku. Počet navzorkovaných hodnot ovlivňuje vzorkovací kmitočet. Vlastnosti digitálního signálu pak můžeme změnit matematickými operacemi.

2.1 Rychlá Fourierova Transformace

Pomocí Rychlé Fourierové transformace *FFT* (*Fast Fourier Transform*) je vypočten Diskrétní Fourierova transformace *DCT* (*Discrete Fourier Transform*). Výsledek je stejný v obou případech, *FFT* je ale mnohem rychlejší. Fourierova transformace vychází z předpokladu, že každý periodický signál lze rozložit na harmonické signály, které jsou různého kmitočtu, fáze a amplitudy. *FFT* je používán pro přesun řečového signálu z časové oblasti do oblasti kmitočtové. Inverzní rychlá Fourierova transformace *IFFT* (*Inverse Fast Fourier Transform*) je používána pro konverzi signálu v kmitočtové oblasti do oblasti časové. [10]

2.2 Melovské frekvenční keprstrální koeficienty

Prvním krokem automatického rozpoznávacího systému řeči je určení komponentů audiosignálu, které jsou užitečné tzn., které obsahují řeč. Ostatní informace např. šum pozadí jsou vyřazeny. Zvuky tvořené lidmi jsou filtrovány tvarem vokálního traktu. Při průchodu zvuku vokálním traktem jsou některé kmitočty tohoto zvuku zesíleny a některé zeslabeny. Vokální trakt tedy představuje filtr. Protože vokální trakt každého člověka má jinou formu, hlas každého jedince bude obsahovat jiné spektrum kmitočtů. Jestliže je určen tvar vokálního traktu přesně, je možné určit i jaký foném byl produkován. Tvar vokálního traktu se projevuje v obálce krátkodobého výkonového spektra. Úkolem Melovských frekvenčních keprstrálních koeficientů *MFCC* (*Mel Frequency Cepstral Coefficients*) je určení právě této obálky s dostatečnou přesností. Kroky implementace jsou následující: [2]

1. v první části je provedena preemfáze, čímž se zesilují vyšší kmitočty signálu,
2. signál je standardně rozdělen do 25 ms rámců s krokem 10 ms, což znamená, že se rámce překrývají. Na rámce je aplikováno Hammingovo okénko, které potlačuje vzorky na krajích okénka.
3. rámce jsou přesunuty do kmitočtové oblasti pomocí *FFT*.

4. je generována Melovská banka filtrů trojúhelníkového tvaru pomocí kterých se vyfiltrují další nežádoucí data. Protože člověk neumí rozpoznat dvě k sobě velmi blízké kmitočty, musí se pomocí těchto filtrů vypočítat suma energie v jednotlivých kmitočtových pásmech. Energie je vypočtena tak, že se vynásobí každý filtr kmitočtovým spektrem signálu. Výsledkem jsou hodnoty *FBE* (*Frequency Band Energy*), které indikují jaká velká je energie v oblasti jednotlivých filtrů.
5. v poslední části se aplikuje *DCT* na logaritmus výstupních hodnot filtrů *FBE* a tím je signál převeden do keprstránní oblasti.

2.3 Dynamické borcení času

Dynamické borcení času *DTW* (*Dynamic Time Warping*) je dlouho známá a používaná technika v oblasti rozpoznání řeči a slouží pro měření podobnosti mezi dvěma časovými signály, které nemusí být stejně dlouhé. Pomocí *DTW* je vypočten optimální rozmezí mezi dvěma časovými sekvencemi podle jistých pravidel.

V případě dvou časových sekvencí, sekvence Q s délkou x , a sekvence R s délkou y je nejprve vytvořena matice $\mathbf{SM}$ o rozměrech (x, y) , což je matice obsahující vzdálenosti vzájemných vzorků a je založena na výpočtu kosinového uhlu mezi vektory Q a R . Výsledná matice bude obsahovat hodnoty od -1 do 1 , kde podobné rámce mají hodnoty blízké 1 , a odlišné hodnoty blízké -1 . V této matici je nutné najít nejlepší cestu, což je umožněno Dynamickým programováním *DP* (*Dynamic Programming*). Cílem *DP* je najít cestu, která bude nejkratší. K dosažení tohoto cíle je vytvořena nová matice, která bude obsahovat minimální vzdálenosti k dosažení specifického bodu z počátečního bodu. Podmínky podle kterých je cesta hledána:

1. cesta musí začít v bodě $(1,1)$ a končit v bodě o souřadnicích (x, y) ,
2. není možné postupovat zpátky v čase, tzn. z libovolného bodu je možné sestupovat doprava, směrem dolů a diagonálně.

Postup *DP* lze matematicky popsat následovně:

$$\gamma(i, j) = d(q_i, r_j) + \min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\}, \quad (2.1)$$

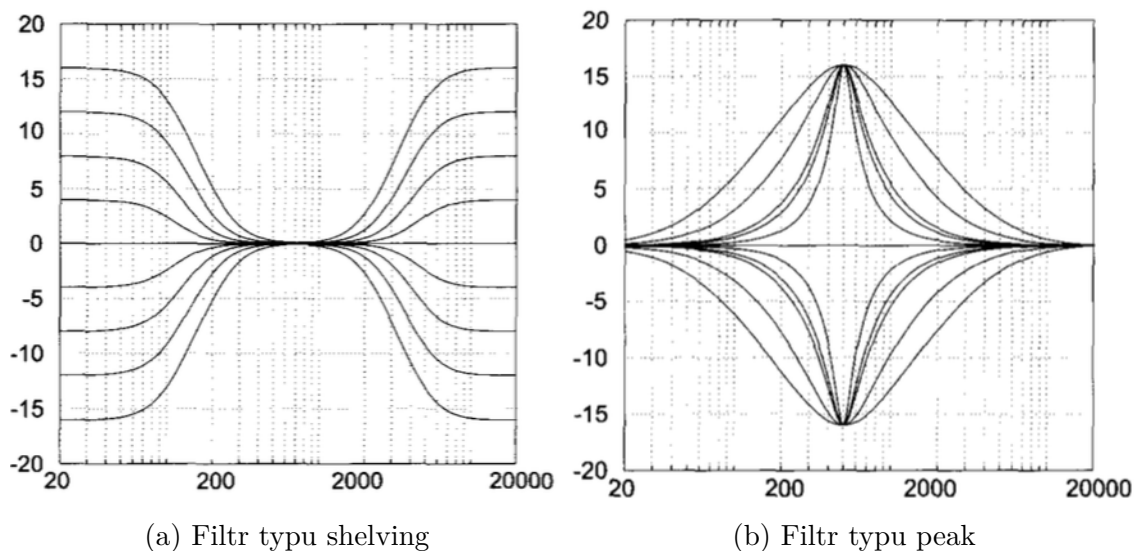
kde $d(q_i, r_j)$ je vzdálenost v aktuální buňce a $\gamma(i, j)$ je souhrnná vzdálenost $d(q_i, r_j)$ a minimální vzdáleností ze tří sousedních buněk.

Podle tohoto vzorce je vypočtena cesta s nejnižší cenou z každého bodu ke každému sousednímu bodu, k čemuž se přičte vzdálenost v aktuální buňce. Po dokončení tohoto algoritmu je vytvořena matice, ve které mají podobající se buňky cenu blízkou k 0 . [2, 6]

2.4 Ekvalizéry

Filtry dolní, horní a pásmové propusti jsou používány pro utlumení kmitočtového spektra pod nebo nad mezním kmitočtem. Na rozdíl od toho, ekvalizéry tvarují určitý část kmitočtového spektra. Ostatní části zůstávají beze změny.

Filtry shelving 2.1a zesilují nebo potlačují nízké nebo vysoké kmitočty podle parametrů mezního kmitočtu f_c a zisku G . Filtry peak 2.1b zesilují nebo potlačují kmitočty kolem středního kmitočtu f_c s šířkou pásma f_b podle zisku G . [17]



Obr. 2.1: Ekvalizéry [17]

2.5 De-identifikace pomocí změnou energií kmitočtových pásem

Cílem této metody je změna hlasu mluvčího na hlas jiného mluvčího. Tato metoda je založena na tom, že výška řečového signálu je závislá na kmitočtových komponentech tohoto signálu. Koeficienty *FBE* a *MFCC* tvoří základní část celého algoritmu, jelikož podle těchto koeficientů jsou určeny nové energie kmitočtových pásem. Protože promluvy nejsou stejně dlouhé, pomocí *DTW* jsou vytvořeny nové matice **FBE**, které jsou stejných rozměrů. Pomocí koeficientů *FBE* původního mluvčího a koeficientů *FBE* jiného mluvčího, jsou vypočteny koeficienty nových hladin energií. Tyto koeficienty říkají jak musí být energie daného kmitočtového pásma změněna, aby se hodnoty energií původního mluvčího shodovaly s energiemi druhého mluvčího. Původní promluva je pak filtrována, filtry typů shelving a peak, kde zisk G odpovídá koeficientům vypočteným v předešlém kroku.

2.6 De-identifikace pomocí změny výšky hlasu

Úkolem této metody je změnit výšku hlasu člověka bez toho, aby se přitom změnila doba trvání promluvy. Je založena na roztažení a zúžení časové domény signálu a následném převzorkování signálu.

V prvním kroku je změněna délka řečového signálu bez změny výšky hlasu. V druhém kroku je naopak signál znova převeden do jeho původní délky, přičemž se změní i výška hlasu. Výška výsledného signálu bude buď hlubší nebo vyšší na základě toho, jestli byl signál roztažen nebo zhuštěn v prvním kroku. Pořadí těchto operací je časově nezávislé. Aby se minimalizoval čas výpočtu, když koeficient $\alpha \leq 1$ je pořadí zúžení a převzorkování a naopak když $\alpha > 1$ je pořadí převzorkování a roztažení. [10]

2.6.1 Změna délky trvání řečového signálu

Algoritmy pro změnu délky trvání signálu jsou použity v případě, kdy je zaznamenána promluva o délce např. 12 s, přičemž maximální povolená délka záznamu je jen 10 s. Namísto toho aby se promluva znova nahrála, může být uměle zkrácena, přičemž nedochází ke ztrátě informací. Úkolem algoritmů pro změny délky časového signálu je prodloužit nebo zkrátit zvukový soubor, který má délku Z na délku $Z' = \alpha * Z$, kde α je koeficient posunu. Faktor $\alpha \leq 1$ znamená zkrácení a $\alpha \geq 1$ znamená prodloužení signálu. Při změně délky se ale frekvenční vlastnosti jako je výška hlasu mluvčího z příkladu výše nesmí změnit. Při změně výšky hlasu by se mluvčí stal nerozpoznatelným.

Základní myšlenka změny délky v čase je rozdělit audiosignál na dílčí segmenty. Pokud chceme signál delší, některé segmenty jsou opakovány a naopak, chceme-li signál kratší, musíme některé segmenty vyřadit. Jedna z možných implementací algoritmu je *SOLA* (*Synchronous Overlap and Add*). [10, 17]

2.6.2 Změna výšky zvuku

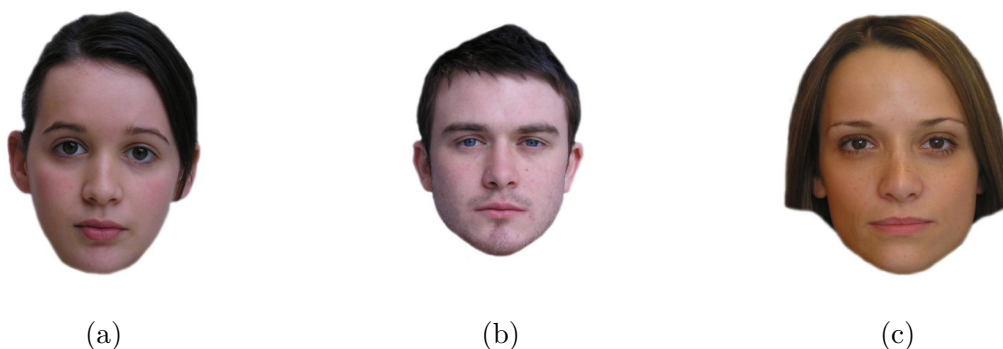
Pomocí algoritmů *FFT* a *IFFT* je možné změnit délku i výšku hlasového signálu, pokud ale jsou kombinovány algoritmem změny časové domén *SOLA*, časová změna *FFT* a *IFFT* je kompenzována. Nejprv je algoritmem *SOLA* změněna délka signálu z délky Z na délku Z' . Algoritmem *IFFT* je změněna výška zvuku a délka, kde délka analýzy *IFFT* je Z/α . [10]

3 REALIZACE PROGRAMU PRO DE-IDENTIFIKACI OBLIČEJE

Následující dva programy byly realizovány v programovém prostředí Matlab. Matlab pracuje s maticemi, každý pixel v obrázku je reprezentován v Matlabu jako prvek matice. Výhoda programovacího jazyku Matlab je, že se obraz P o rozměrech $m_1 \times m_2$ přeloží na matici $\mathbf{P}$ o rozměrech $M_1 \times M_2$, tímto způsobem se zjednodušila realizace programu.

První program byl použit na předzpracování obrazů. Předzpracování spočívá v určení hodnot potřebných pro další zpracování a také v usnadnění další práce s obrazy. Výsledky předzpracování jsou textové soubory obsahující souřadnice hlavních rysů obličeje a přejmenované soubory černobílých obrazů obličeje. Druhý program byl použit pro samotnou de-identifikaci. V návrhu byla věnována pozornost tomu, aby se de-identifikovala jenom oblast obsahující obličej.

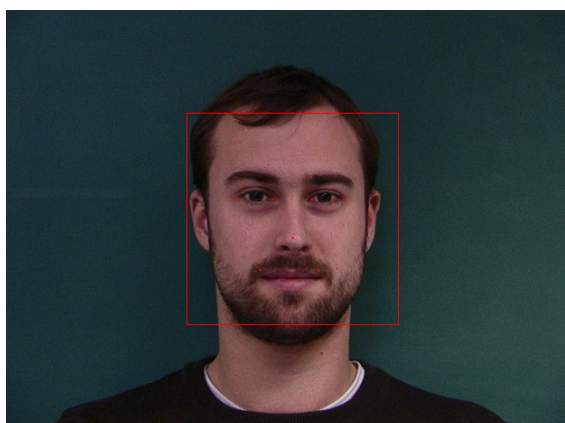
Pro ověření funkcí programu de-identifikace byly použity obrazy obličeje z databáze *GUFD* (*Glasgow Unfamiliar Face Database*) [1] a další obrazy z databáze *IMM* (*Informatics and Mathematical Modelling*) [11], které byly předzpracovány prvním programem.



Obr. 3.1: Obrazy z databáze *GUFD* [1]

3.1 Program - Předzpracování

Program začíná načtením obrazu do proměnné *obr*, který je pak převeden na černobílý obraz. V dalším kroku spustitelný soubor „findFace” určí souřadnice krajních bodů očí, krajních bodů úst, dvou bodů na špičce nosu a souřadnice čtverce obsahující obličej. Souřadnice určují polohu pixelů a ukládají se do textového souboru „position”. Po úspěšném nalezení tváře a určení souřadnic hlavních rysů, „findFace” vrací hodnotu 1, když tato hodnota není 1, objeví se chybové hlášení a program se ukončí.



Obr. 3.2: Vypočtené souřadnice

V další části programu je obraz a příslušný textový soubor přejmenován. V následujícím kroku je obraz a textový soubor vložen do složky „Obrazy” resp. „Pozice”. Algoritmus je v cyklu *for*, což zaručuje, že každý obraz bude předzpracován stejným způsobem.

3.2 Program - De-identifikace obličeje

Program začíná výběrem obrazu, který chceme de-identifikovat. Z příslušného textového souboru jsou načteny souřadnice, do proměnných x, y, h, w a *point*, kde x je začátek čtverce obsahující obličej na x-ové ose, y je začátek čtverce na y-ové ose, w je šířka a h je výška čtverce. Do proměnné *point* jsou uloženy souřadnice bodů: očí, úst a nosu. Po načtení obrazu se objeví menu s výběrem patnácti metod de-identifikace, nového obrazu a ukončení programu, objeví se také vybraný obraz.

Hlavní část programu je součástí funkce *while*, tzn. po aplikaci vybrané metody de-identifikace na obličej se znovu objeví menu s výběrem metod. Obraz se nejprve načte do proměnné *vstup* a pak až v cyklu *while* do proměnné *výstup*. Obsah proměnné *vstup*, tj. původní obraz, je pak po vykonání vybrané metody překopírován do proměnné *výstup*. Podprogramy pak pracují s proměnnou *výstup*, *vstup* zůstane beze

změny. Tato skutečnost zajišťuje, že každý podprogram pracuje s původním obrazem a umožňuje tak aplikaci různých metod de-identifikace na stejný obraz obličeje, aniž by program musel být ukončen a znovu spuštěn. Po samotné de-identifikace obličeje se objeví vedle původního obrazu i de-identifikovaný obraz.

Protože podprogramy Rozmazání, Pixelace a k -Same vyžadují informaci o míře de-identifikace, po volání těchto metod se objeví další okno žádající číselnou hodnotu. Po volání metod Zatemnění, Bar mask nebo Mouth only se také objeví okno žádající číselnou hodnotu, ale v tomto případě se nastavuje jenom barva obdélníků, např. v podprogramu Bar mask je možné nastavit barvu pásu zakrývajícím oči. Tyto hodnoty jsou uloženy do proměnné *numb*. Ostatní podprogramy nevyžadují zadání dalších hodnot.

Zatemnění, Bar mask, T mask, Mouth Only

Pomocí souřadnic z textového souboru je definována oblast, ve které se změní barva obrazu. V případě podprogramu Zatemnění hodnota veškerých pixelů v oblasti čtverce obsahující obličej je nastavena na zadanou hodnotu. Výchozí hodnota pixelů v oblasti je vždy 0, což je černá barva. Námi nastavená hodnota odstínu se ukládá do proměnné *numb*. Další podprogramy pracují na stejném principu, ale oblast pokrytí je jiná. V případě podprogramu Bar mask se změní hodnota pixelů obdélníku obsahující oči a oblast spojující oči. Syntaxe tohoto podprogramu je znázorněna ve výpisu 3.1. Souřadnicemi *point*(6, 2), *point*(7, 2), *point*(6, 1) a *point*(7, 1) jsou určeny krajní body očí. V podprogramu T mask je tato oblast rozšířena o další obdélník obsahující nos. V podprogramu Mouth Only je definován obdélník, který pokrývá oblast od nosu, až po horní okraji hlavy.

```
214 case{9}
215     if (numb>0)
216         output (point (6,2)-20:point (7,2)+20,...
217                 point (6,1)-25:point (7,1)+25)=numb;
218     else
219         output (point (6,2)-20:point (7,2)+20,...
220                 point (6,1)-25:point (7,1)+25)=0;
221     end
```

Výpis kódu 3.1: Podprogram Bar mask

Průměrování

V prvním kroku se vygeneruje náhodné číslo, od jedné do celkového počtu obrazů, různé od pořadového čísla námi zvoleného obrazu. Načte se obraz s náhodně generovaným pořadovým číslem a příslušný textový soubor, poté se vypočítá průměrný obraz těchto dvou obrazů podle rovnice:

$$Average(\mathbf{P}_1, \mathbf{P}_2) = \frac{1}{2} \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} \mathbf{P}_1(i, j) + \mathbf{P}_2(i, j), \quad (3.1)$$

kde i je řádek a j je sloupec, M_1 počet řádků a M_2 počet sloupců, $\mathbf{P}_1$ a $\mathbf{P}_2$ jsou obrazy obličejů. Protože obrazy obličejů jsou různých rozměrů, souřadnice druhého obrazu je třeba aproximovat. Proměnné o a s jsou tyto nové souřadnice. Část průměrování je vidět ve výpisu 3.2, kde pr je náhodně vygenerovaný obraz, kterým je zprůměrován námi vybraný obraz obličejů. xx , yy , ww , hh jsou souřadnice čtverce obsahující obličej pr ; i , o souřadnice řádků; j , s souřadnice sloupců.

```
106     for i=y:y+h
107         o=round((i*(hh+yy))/(h+y));
108         for j=x:x+w
109             s=round((j*(ww+xx))/(w+x));
110             output(i,j)=(output(i,j)+pr(o,s))/2;
111         end
112     end
```

Výpis kódu 3.2: Část podprogramu Average

Pixelace

V oblasti obličejů jsou dílčí kostky pixelů o rozměrech $p \times p$ zprůměrovány, a následně jsou touto hodnotou pixely ve stejné kostce nahrazeny. Proměnnou p je nastavena velikost jednotlivých kostek. Algoritmus je vidět ve výpisu 3.3.

```
117 case {3}
118     p=numb;
119     for i=y:p:y+h
120         for j=x:p:x+w
121             output(i:(i+p-1),j:(j+p-1))=ones(p).*mean...
122                 (mean(output(i:(i+p-1),j:(j+p-1)))));
123         end
124     end
```

Výpis kódu 3.3: Podprogram Pixelace

Rozmazání

V prvním kroku je vytvořena nová matice o rozměrech čtverce obsahující obličej. Využitím speciální funkce Matlabu *fspecial*, je vytvořena matice obsahující Gaussova filtru. Rozmazaný obraz je vypočten podle vzorce:

$$G_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (3.2)$$

pomocí funkce *imfilter*, která slouží pro 2D konvoluci. Hodnota b ve výpisu 3.4 odpovídá hodnotě σ ve vzorci 3.2. Proměnnou b je nastavena míra rozmazání obrazu.

```
134 case {4}
135     b=numb; R=zeros(h-y,w-x);
136     u=fspecial('gaussian', size(R), b);
137     output=imfilter(output, u);
```

Výpis kódu 3.4: Podprogram Rozmazání

Negative, Random Noise a Random Noise Černobílý

V metodě Negative je vypočtena převrácená hodnota každého pixelu $255 - x$, v rámci čtverce obsahující obličej, jak to vyplývá z výpisu 3.5. V prvním kroku metody Random Noise je vygenerována matice náhodných hodnot, které jsou zaokrouhleny na 1 nebo 0. Pokud v náhodné matici je hodnota buňky x rovna 1, hodnota pixelu v obrazu v bodě x je vyměněna s náhodně vygenerovanou hodnotou mezi 0 a 255. Podprogram Random Noise je vidět ve výpisu 3.6. Random Noise Černobílý se liší od metody Random Noise tím, že hodnota pixelu x je vyměněna náhodným číslem buď 0 nebo 255.

```
151         case {5}
152             for i=y:y+h
153                 for j=x:x+w
154                     output(i,j)=255-output(i,j);
155                 end
156             end
```

Výpis kódu 3.5: Podprogram Negative

```

185         case {7}
186             [m,n]=size(output); Rn=round(rand(m,n));
187             for i=y:y+h
188                 for j=x:x+w
189                     if Rn(i,j)==1
190                         output(i,j)=randi(255,1,1);
191                     end
192                 end
193             end

```

Výpis kódu 3.6: Podprogram Random Noise

Threshold a Ordinal

V metodě Threshold je podle dvou podmínek a v závislosti na zvoleném čísle t nastavena hodnota každého pixelu. Podmínka říká, že pokud je hodnota pixelu x menší než prahová hodnota t , hodnota pixelu se nastaví na 0, což je černá barva a naopak když hodnota pixelu x je větší než t , hodnota pixelu se nastaví na 255, což se projevuje jako bílá barva. Algoritmus lze vidět ve výpisu 3.7. Výsledkem této metody je obraz obličeje černé a bílé barvy. V metodě Ordinal, podle pěti předem definovaných podmínek je výsledná hodnota pixelu nastavena na jednu z pěti předem definovaných hodnot.

```

201         case {8}
202             t=numb;
203             for i=y:y+h
204                 for j=x:x+w
205                     if output(i,j)<=t
206                         output(i,j)=0;
207                     elseif output(i,j)>t
208                         output(i,j)=255;
209                     end
210                 end
211             end

```

Výpis kódu 3.7: Podprogram Threshold

Mouth Only Černobílý a Negative Černobílý

Tyto metody kombinují Mouth Only a Negative s metodou Threshold. Programy se skládají ze dvou fází. V první fázi je aplikována na obraz metoda Threshold pak příslušný podprogram tj. Mouth Only nebo Negative.

k-Same

V první části podprogramu jsou vypočteny eukleidovské vzdálenosti mezi vybraným obrazem a mezi obrazy ve vstupní sadě obrazů podle vzorce:

$$euclid(\mathbf{P}_1, \mathbf{P}_2) = \sqrt{\sum_{i=1}^{M_i} \sum_{j=1}^{M_j} |\mathbf{P}_1[i, j] - \mathbf{P}_2[i, j]|^2}, \quad (3.3)$$

kde i je řádek a j je sloupec, M_1 počet řádků a M_2 počet sloupců, $\mathbf{P}_1$ a $\mathbf{P}_2$ jsou obrazy obličejů. Výsledné vzdálenosti se ukládají do proměnné *os*. Po výpočtu vzdáleností mezi všemi obrazy jsou tyto hodnoty uspořádané podle velikosti vzestupně. Proměnná k určuje kolik obrazů bude v dalším kroku zprůměrováno.

V druhé části programu je vypočten průměrný obraz podobným způsobem jak v podprogramu Průměrování a protože obrazy nejsou stejných rozměrů, souřadnice ostatních obrazů jsou aproximovány. Rozdíl mezi metodou k-Same a Průměrování spočívá ve výběru obrazů, se kterými bude původní obraz průměrován. V metodě Průměrování je vybrán druhý obraz náhodně a v případě metody k-Same podle eukleidovské vzdálenosti. V této části programu je podle proměnné *os* určen k nejbližších obrazů a následně jsou tyto obrazy zprůměrovány původním obrazem.

Nový obraz

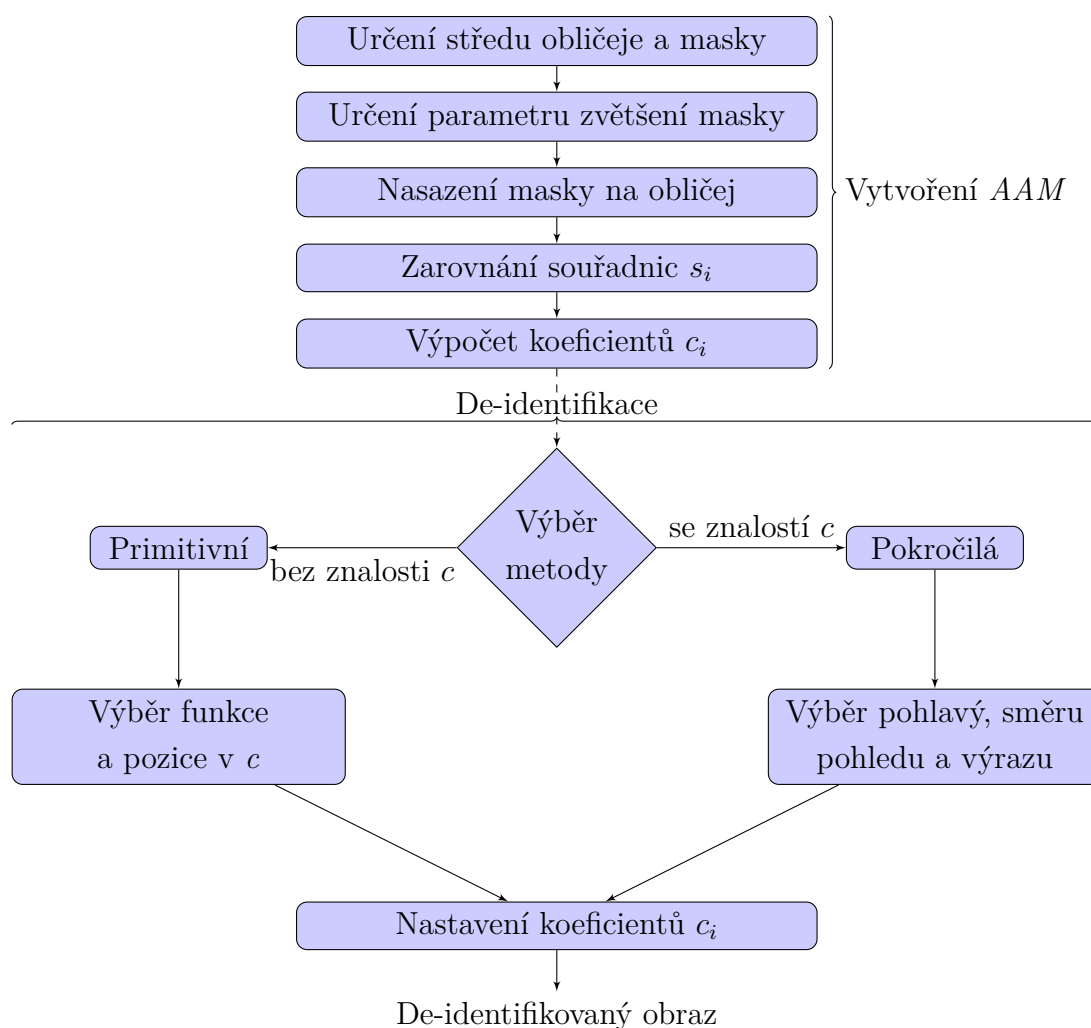
V tomto podprogramu jsou vymazány všechny proměnné, aktivní okna jsou zavřena a je otevřeno nové okno pro výběr obrazu. Jedná se vlastně o restartování hlavního programu.

Ukončit

Hlavní program je zacyklen v cyklu *while*, dokud proměnná *konec* je rovna 0. Podprogram zavře všechna aktivní okna, nastaví hodnotu proměnné *konec* na 1 a tím je program ukončen.

4 NÁVRH METODY PRO REVERZNÍ DE-IDENTIFIKACI OBLIČEJE

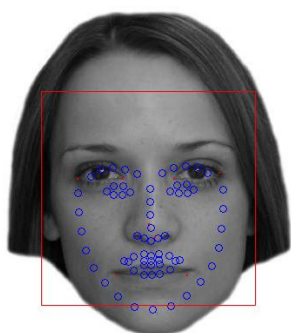
Algoritmus reverzní de-identifikace vychází z poznatků popsaných v části 1.4. Algoritmus lze rozdělit do dvou dílčích ale na sobě navazujících částí. Výstupem první části je parametr vzhledu c . Ve druhé části je tento vektor použit pro de-identifikaci obličeje, která vychází z předpokladu, že výsledný parametr c je lineární kombinací jednotlivých koeficientů c_i . Algoritmus je reverzní, protože původní parametr c je uložen. Tzn. po de-identifikaci je možné rekonstruovat původní obličej pomocí tohoto nezměněného parametru. Blokové schéma algoritmu je vidět na obrazu 4.1.



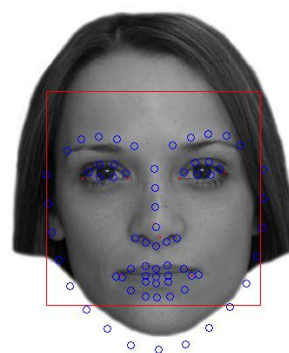
Obr. 4.1: Blokové schéma reverzní metody de-identifikace

4.1 Vytvoření *AAM*

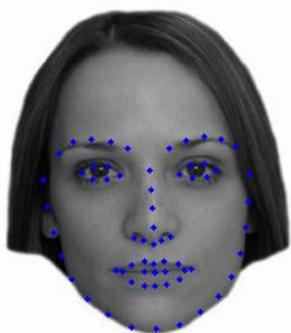
Prvním úkolem je přizpůsobit základní parametr vzhledu s_0 k obličeji, čehož souřadnice jsou předem definované. Přesná pozice a velikost obličeje je určena spustitelným souborem „findFace“. Souřadnice určující střed masky jsou nastaveny tak, aby odpovídaly souřadnici nosu, získaného pomocí „findFace“. Může nastat situace, kdy maska a obličej nejsou stejné velikosti. V případě když je maska menší než obličej, musí být maska zvětšena, viz obraz 4.2a. V prvním kroku je vypočtena hodnota, která říká, kolikrát musí být maska zvětšena, aby pozice mezi očima v obrazu odpovídala pozici mezi očima masky. Dále je vypočtena hodnota, která říká, kolikrát musí být maska zvětšena, aby pozice mezi krajními body rtů v obrazu odpovídala pozici mezi krajními body rtů masky. Výsledná hodnota koeficientu zvětšení je součtem těchto dvou hodnot. Výsledná velikost masky je vidět na obrazu 4.2b. Jelikož souřadnice masky jsou další částí algoritmu doladěny tak, aby byla maska perfektně nasazena na obličej, viz obraz 4.2c, v této části je velikost masky nastavena jen na přibližnou hodnotu.



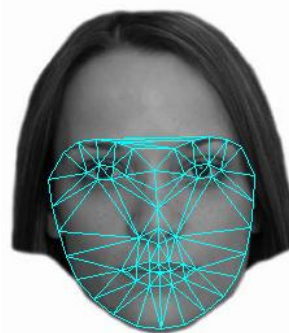
(a) Původní velikost masky



(b) Zvětšená maska



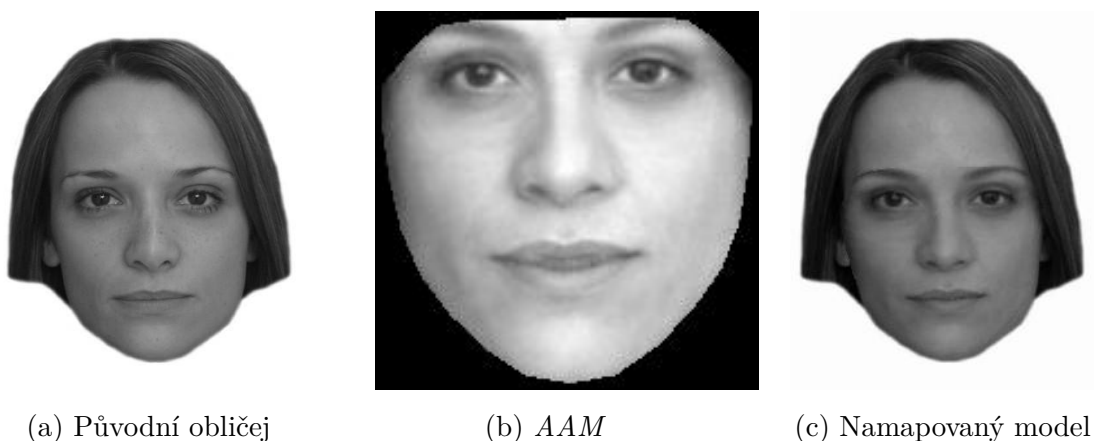
(c) Maska po doladění



(d) Trojúhelníková maska

Obr. 4.2: Nastavení velikosti masky a vytvoření *ASM*

Dále je vypočten vektor c , který je parametrem vzhledu, a nastavuje texturu, tzn. intenzitu veškerých pixelů v oblasti masky. Podle vzorců 1.2 a 1.3 je generován tvar s tj. ASM a vzhled A tj. AAM viz obraz 4.3b. Výsledný model AAM je vytvořen namapováním vzhledu A se základním tvarem s_0 na výsledný tvar s . ASM je rozdělen na trojúhelníky spojením souřadnic, viz obraz 4.2d. Pro každý trojúhelník v s_0 existuje odpovídající trojúhelník v s . Pixely vzhledu A se namapují do tvaru s následujícím způsobem: pro každý pixel x je nejprve zjištěno v kterém trojúhelníku leží, pak je tento pixel přemístěn z trojúhelníku v s_0 do odpovídajícího trojúhelníku v s . Mezi původním obrazem 4.3a a výsledným namapovaným modelem AAM 4.3c je rozdíl téměř neviditelný. Program realizující tuto část algoritmu byl realizován Georgisem Tzimiropoulosem [14].



Obr. 4.3: Rozdíly mezi původním obrazem a modelem AAM

AAM jsou určeny primárně pro napodobení obličeje, ale v další části budou parametry vzhledu c využity pro de-identifikaci. Cílem de-identifikace pomocí parametru c je vytvoření metody, která generuje přirozené obrazy obličeje, tzn. zachovává rysy v obličeji (oči, ústa, nos). Základní myšlenkou těchto metod je změnit koeficienty parametru vzhledu tak, aby výsledný obličej nebyl rozpoznatelný. Tyto metody můžeme rozdělit na primitivní, tj. bez jakéhokoliv předešlé znalosti c (např. aplikace funkcí na vybrané části vektoru) a na pokročilé, které vyžadují znalost vlastností parametru c (změna koeficientů c podle jistých pravidel).

4.2 Reverzní „primitivní” de-identifikace obličeje

Aplikací funkce exponenciály na parametr c odpovídá metodě Zatemnění s rozdílem, že zde je odstraněna jenom oblast obsahující obličej, viz obraz 4.4a. Z toho vyplývá, že nastavením velkých hodnot v parametru c je obraz efektivně de-identifikován, avšak tato metoda nezanechává žádné informace o subjektu.

Aplikací funkce sinus na obrazy je obličej subjektu změněn, zde se ale jedná o metodu která zachovává informace v obličeji, viz obraz 4.4c. Vhodnou změnou prvních několika koeficientů je možné vytvořit obličej, který bude de-identifikován, přičemž zachovává v obličeji veškeré informace. V předchozích příkladech bylo modifikováno jen několik prvních koeficientů ve vektoru c . Na obrazu 4.4d lze vidět obličej, kde funkce sinus byla aplikována jen na druhou polovinu parametru c . Obličej lze bez jakýchkoliv těžkostí rozpoznat už na první pohled. Aplikací funkcí cosinus a logaritmus na tuto část parametru je dosaženo stejného výsledku, z čehož vyplývá, že tato část parametru má malý vliv na výsledný AAM . Aplikací funkce exp na druhou polovinu parametru c , je dosaženo vysokých hodnot buněk, což zapříčiní nerozpoznatelné zkreslení obličeje, viz obraz 4.4b.



(a) $exp[c(1:50)]$



(b) $exp[c(100:200)]$



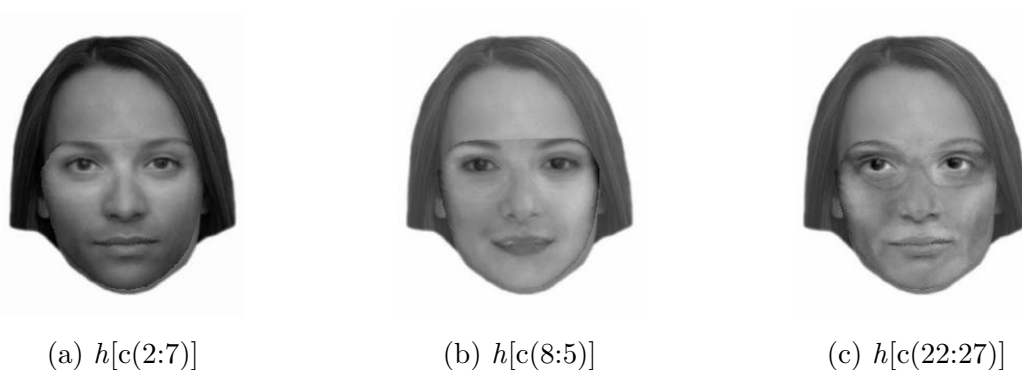
(c) $sin[c(1:50)]$



(d) $sin[c(100:200)]$

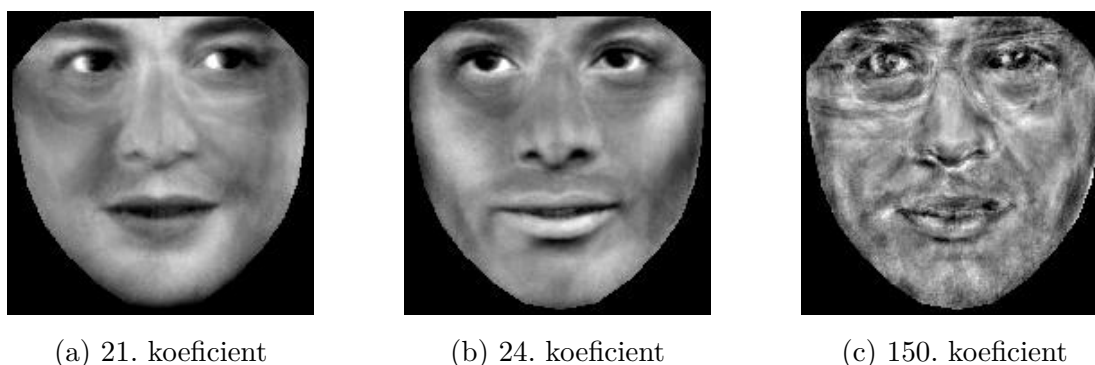
Obr. 4.4: Aplikace funkcí exp a sin na parametr vzhledu c

Aplikace Hammingova okénka na parametr c je založena na poznatcích popsaných v předešlé části této podkapitoly. Hammingovo okénko h je vektor určité velikosti, ve kterém je hodnota uprostřed rovna 1, směrem k okrajům vektoru se tato hodnota postupně snižuje. Vektor h je vynásoben 2000 a následně libovolná část v c je nahrazena tímto vektorem. Ostatní koeficienty c nejsou ovlivňovány. Vynásobením h příliš malým číslem nevede k viditelné změně vzhledu a vynásobením příliš velkým číslem vede ke ztrátě přirozenosti obličeje. Na obrazech 4.5 jsou výsledné modely po aplikaci Hammingova okénka. Hammingovým okénkem délky 5 byly nahrazeny části v první čtvrtině vektoru c . Modely zachovají informace obličeje, jsou přirozeného vzhledu. Hammingovo okénko je efektivní metoda de-identifikace. Nedostatkem této metody je náhodné nahrazení částí c , tzn. výsledný vzhled obličeje nelze nijak jinak ovlivnit.



Obr. 4.5: Aplikace Hammingova okénka na parametr c

4.3 Reverzní „pokročilá” de-identifikace obličeje



Obr. 4.6: Skutečný obsah buněk parametru c

Největší vliv na výsledný model vzhledu mají koeficienty přibližně na prvních 50 místech. Pokud budou veškeré koeficienty vektoru c nastaveny na nulu, a postupně bude po jednom nastaven každý koeficient na hodnotu několika tisíc, bude pozorovatelné, že každá buňka c reprezentuje obličej jiného muže nebo ženy.

Prvních přibližně 50 obličejů budou přirozené např. obličej muže 4.6a a ženy 4.6b, se zvětšující se pozicí v parametru c se postupně zhoršuje kvalita obličeje, viz obraz 4.6c. Dalším cílem je navrhnout metodu de-identifikace, se kterou lze, podle získaných znalostí z analýzy, ovlivňovat výsledný vzhled podle pohlaví, tvaru očí a úst. Ze získaných obrázků viditelných na obrazech 4.6 byly zjištěny vlastnosti parametru c a následně rozděleny. V tabulce 4.1 je uvedeno prvních padesát koeficientů rozdělených podle pohlaví, směru pohledu, výrazu a barvy jejich brýlí. Podle těchto vlastností pak koeficienty byly rozděleny dále do skupin, např. Muž který se dívá dopředu a usmívá se (1, 2, 4, 12, 27), žena která se dívá doleva a má vážný výraz (6, 10) atd. Koeficient, jehož hodnota je vysoká, má na výsledný model větší vliv, naopak koeficient s malou hodnotou má na výsledný model vliv malý. Na základě výše popsaných vlastností je založena metoda, pomocí které je možné nastavovat vlastnosti výstupního modelu.

Tab. 4.1: Rozdělení prvních 50 koeficientů parametru c podle vizuálních vlastností

Dělení podle pohlaví:	
Muži:	1, 2, 4, 5, 7, 12, 15, 17, 18, 19, 21, 23, 27, 28, 29, 30, 32, 33, 34, 36, 37, 38, 41, 42, 44, 45, 47, 48, 50
Ženy:	3, 6, 8, 9, 10, 11, 13, 14, 16, 20, 22, 24, 25, 26, 31, 35, 39, 40, 43, 46, 49
Dělení podle směru pohledu:	
Dopředu:	1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 25, 27, 29, 30, 32, 33, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 49, 50
Doprava:	15, 18, 19, 20, 21, 26, 48
Doleva:	6, 10, 14, 16, 17, 23, 31, 34
Nahoru:	22, 24, 28
Dělení podle výrazu:	
Úsměv:	1, 2, 4, 6, 10, 12, 21, 27, 31, 35, 36, 41, 42, 44, 45, 48, 50
Vážný:	8, 9, 15, 16, 17, 18, 19, 22, 23, 24, 25, 26, 28, 29, 30, 32, 33, 34, 37, 38, 40, 43, 46, 47
Dělení podle barvy brýlí:	
Bílé:	33, 38, 40
Černé:	25, 26, 35, 41, 43, 49

Program de-identifikace byl realizován v Matlabu. V případě primitivního přístupu je možné vybrat z několika funkcí ($\sin$, $\cos$, $\exp$, $\ln$, x^n) a dále pozici nebo rozsah koeficientů v c , na které bude aplikována vybraná funkce. Nevýhodou těchto metod je, že není možné ovlivnit vlastnosti výsledného modelu. V pokročilém přístupu je tato nevýhoda eliminována, zde je možné vizuální vlastnosti výsledného modelu předem nastavit. Na modelu 4.7a je muž dívající se doleva a neusmívá se. Ke správnému výsledku se lze dopracovat následovně: V prvním kroku jsou koeficienty žen (3, 6, 8, 9, 10, 11) potlačeny, tj. nastaveny na 0. Pak koeficienty mužů (1, 2, 4, 5, 7, 12) ty jsou zvýšeny na 1000. V dalším kroku jsou koeficienty mužů dívající se doleva s vážným výrazem (17, 23, 34) nastaveny na hodnotu 700. V posledním kroku jsou koeficienty mužů s vážným výrazem. Ostatní koeficienty nejsou ovlivněny.

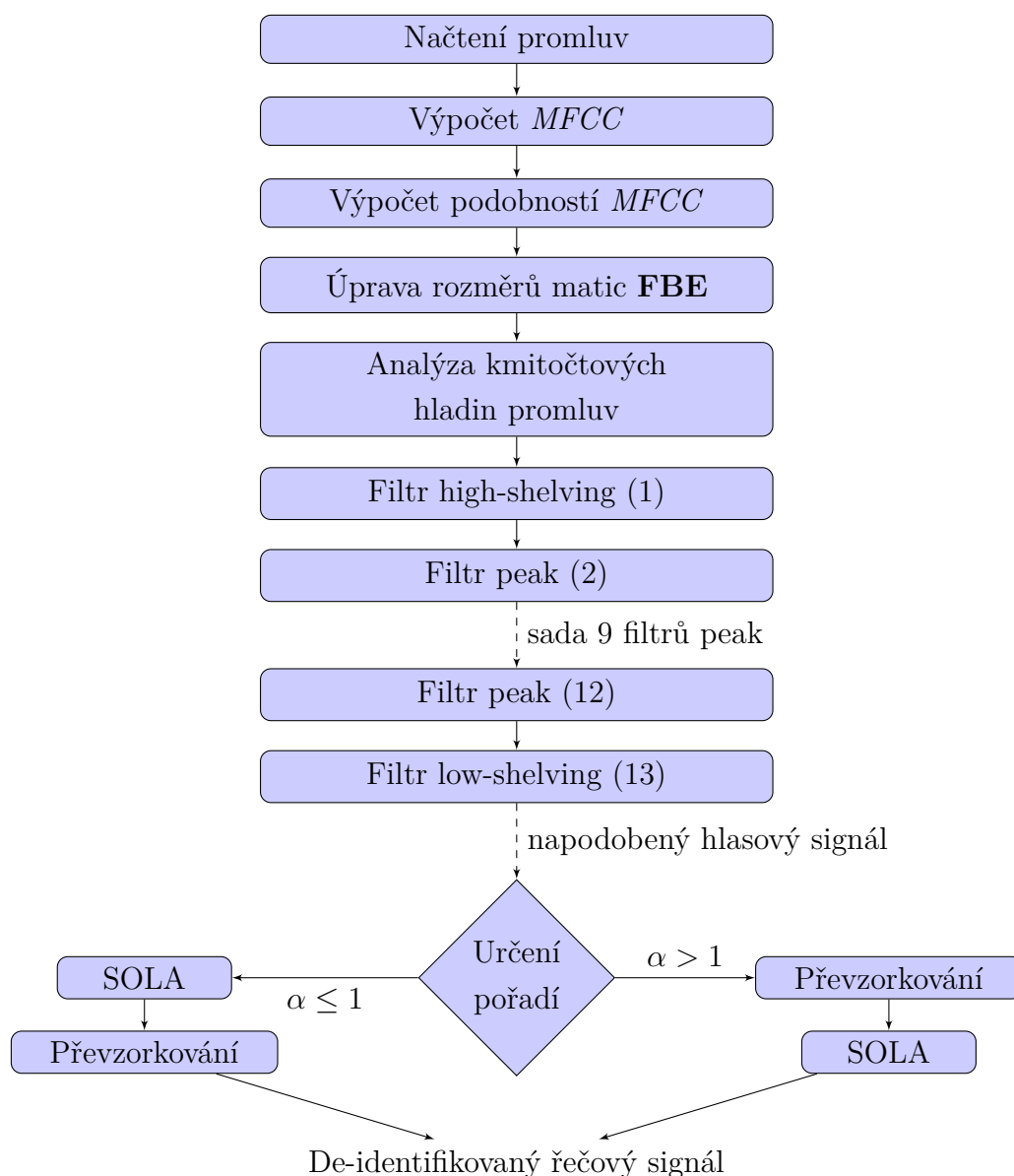
Hodnoty jednotlivých koeficientů lze dále doladit, ale vlastnosti jednotlivých koeficientů se přitom nemění. Tzn. dva stejné vektory c aplikovány na různé obličeje změni každý obličej stejným způsobem. AAM 4.7a a 4.7d, 4.7b a 4.7e, 4.7c a 4.7f byly vytvořeny pomocí stejného parametru vzhledu. Rozdíl mezi obrazy 4.7a a 4.7b (4.7d a 4.7e) je v tom, že v případě 4.7a (4.7d) jde o model muže, který se neusmívá a v případě 4.7b (4.7e), který se usmívá. Ve skutečnosti rozdíl mezi těmito obrazy nelze vidět, přestože byly nastaveny různé hodnoty jejich koeficientů, tzn. pomocí AAM nelze změnit výraz obličeje pouze pomocí ASM.



Obr. 4.7: Výsledné de-identifikované obrazy obličeje

5 NÁVRH METODY PRO DE-IDENTIFIKACI HLASU

Algoritmus vychází z předpokladu, že řeč se skládá z různých kmitočtů, které mají různé hodnoty intenzit. Hlas jedince tedy závisí na tom, na jakých kmitočtech mluví a na tom, jakou mají energii jednotlivé kmitočty. Výstupem první části algoritmu je napodobený hlasový signál tj. kmitočtové vlastnosti zvolené promluvy jsou upraveny podle kmitočtových vlastností jiné promluvy. Po aplikaci dalších uprav je výstupní signál de-identifikovaný. Blokové schéma algoritmu je vidět na obrazu 5.1.

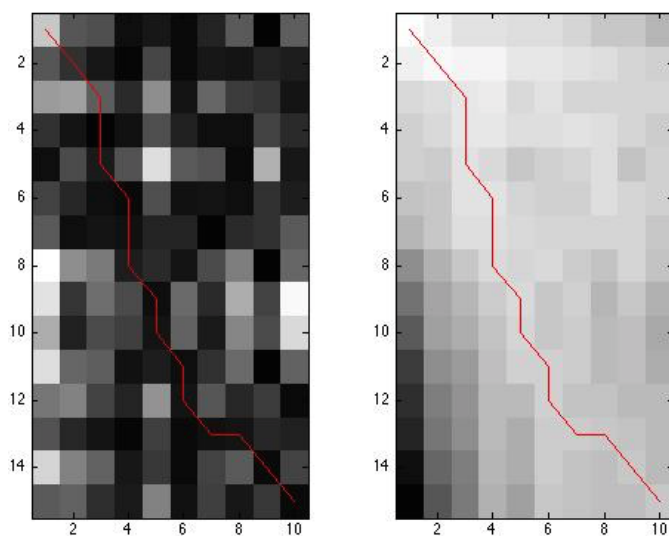


Obr. 5.1: Blokové schéma de-identifikace hlasu

5.1 Napodobení hlasu jiného mluvčího

Algoritmus začíná načtením promluv pak následuje výpočet prvních dvanácti, včetně nultého *MFCC*. V této části jsou vypočteny *MFCC* každé vstupné promluvy, podle postupu v části 2.2. Pro další práci s maticemi **FBE** je nutné, aby byly stejných rozměrů. Problémem ale je, že promluvy nejsou stejně dlouhé a tak ani matice obsahující koeficienty *FBE* nejsou stejných rozměrů. Z tohoto důvodu je použit *DTW*.

Podobnost mezi dvěma maticemi *MFCC*, je založena na výpočtu kosinové vzdálenosti dvou matic. Ve výsledné matici podobné rámce mají hodnoty blízké 1, a odlišné hodnoty blíží se -1 . Na levé straně obrazu 5.2 je tato matice znázorněna graficky, černé čtverce jsou blízké hodnoty k 1, bílé jsou blízké k hodnotě -1 .

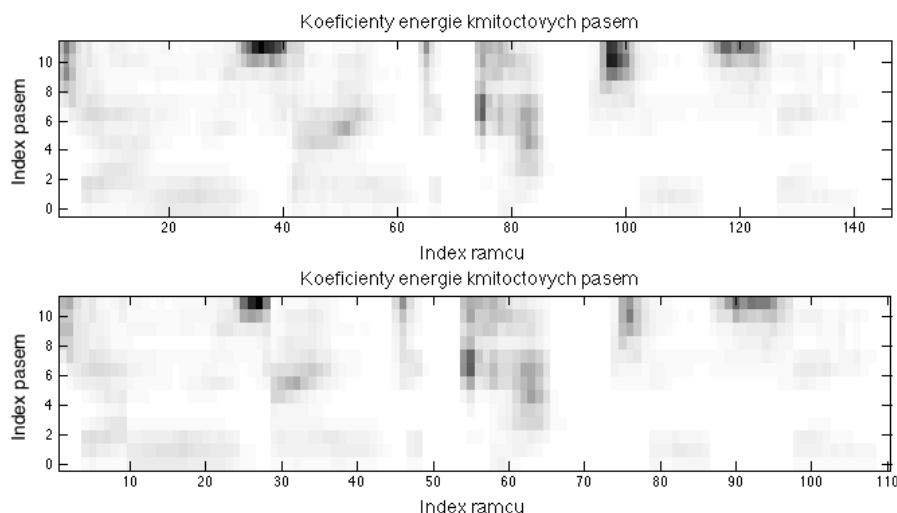


Obr. 5.2: Ukázka nejkratší cesty mezi dvě matice

Algoritmus úpravy rozměrů lze vysvětlit následujícím příkladem: matice **A** je o rozměrech 3×15 a matice **B** o rozměrech 3×10 . Cílem je upravit matici **A** tak, aby výsledná matice **A_n** byla o rozměrech 3×10 , s tím, že bude co nejvíce podobná matici **B**. V prvním kroku je vypočtena matice podobnosti **SM**, která bude o rozměrech 15×10 . Každá buňka této matice reprezentuje vzdálenost mezi jednotlivými prvky matic **A** a **B**. V další části je vypočtena matice, která již obsahuje ceny dostat se z počátku (1,1) matice **SM** do libovolného bodu (i, j) . Výsledná matice je na pravé straně obrazu 5.2 a nejkratší cesta je znázorněna červenou čarou, která dodržuje výše popsané podmínky.

V ideálním případě by cesta z levého horního rohu do pravého dolního rohu byla diagonální, ale protože počet řádků a sloupců v matici není rovný, není možné toho dosáhnout. Cesta v matici obsahuje vertikální a horizontální úseky, tzn. tyto části

cesty musí být odstraněny. Z obrazu 5.2 je zřejmé, že pokud by byla odstraněna vertikální část cesty, výsledkem by byla diagonála. Může ale nastat situace kdy vertikálních částí je větší počet než $i - j$ kde i je počet řádků a j je počet sloupců matice $\mathbf{SM}$, v takových případech se v cestě vyskytují i horizontální úseky. V obrazu je vidět, že vertikálních částí je šest, tzn. šest sloupců z matice $\mathbf{A}$ se musí odstranit. Šest je o jeden víc než $i - j$. Proto aby bylo dosaženo správného výsledku, musí být tento sloupec matice $\mathbf{A}$ zduplikován.

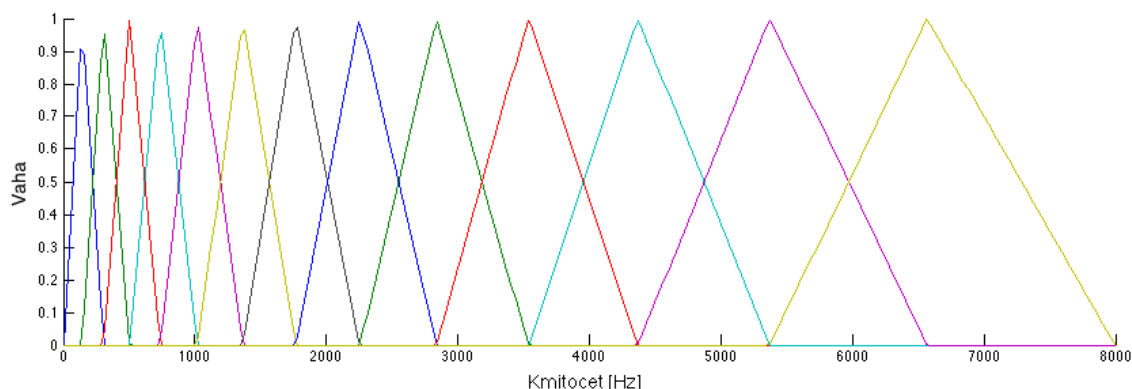


Obr. 5.3: Matice obsahující FBE původní a nové délky

Na obrazu 5.3 je vidět matici obsahující koeficienty FBE_1 původního mluvčího nahoře a pod ní je matice výsledná s koeficienty $nFBE_1$. Nová matice má stejné rozměry jako matice s koeficienty FBE_2 dalšího mluvčího, kde jsou už odstraněny a zduplikovány příslušné sloupce první matice podle výše popsaného postupu.

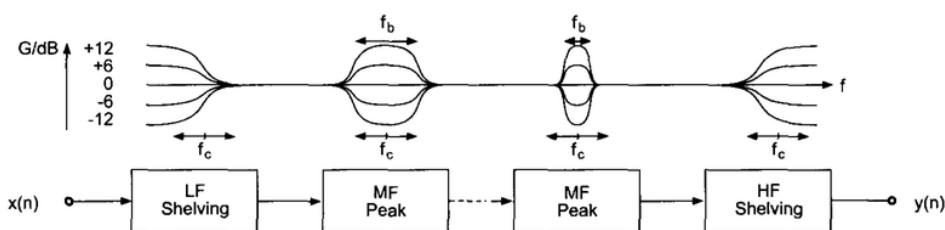
Teď už jsou matice $nFBE_1$ a FBE_2 stejných rozměrů. Výpočet koeficientů podle kterých jsou změněny energie v jednotlivých kmitočtových pásmech začíná podělením matice $nFBE_1$ maticí FBE_2 prvek po prvku, pak je vypočtena průměrná hodnota každého řádku výsledné matice. Pokud by se tato matice vynásobila, každým sloupcem matice FBE_2 , výsledná matice by se zhruba podobala matici $nFBE_1$. Dalším upravením výsledných hodnot této matice je určeno kolikrát se musí zesílit nebo potlačit kmitočtové pásmo promluvy 1 aby bylo dosaženo kmitočtových vlastností promluvy 2.

V dalším kroku je vygenerována matice obsahující kmitočtové vlastnosti 13 filtrů od 0 Hz do 8000 Hz, dále je vygenerován vektor f , který obsahuje body kmitočtové hodnoty počínaje 0 Hz. První filtr začíná v prvním bodě, dosáhne vrcholu v bodě druhém a vrátí se do nuly v bodě třetím. Druhý filtr začíná v bodě druhém, dosáhne vrcholu v bodě třetím a vrátí se do nuly ve bodě čtvrtém. Filtry se překrývají.



Obr. 5.4: Vygenerovaná melovská banka 13 filtrů

Rozložení filtrů je vidět na obrazu 5.4. Z vektoru f jsou určeny šířky jednotlivých kmitočtových pásem. Dále je seřazena sada 13 filtrů, kterými je pak promluva filtrována, jak to je vidět na obrazu 5.5.

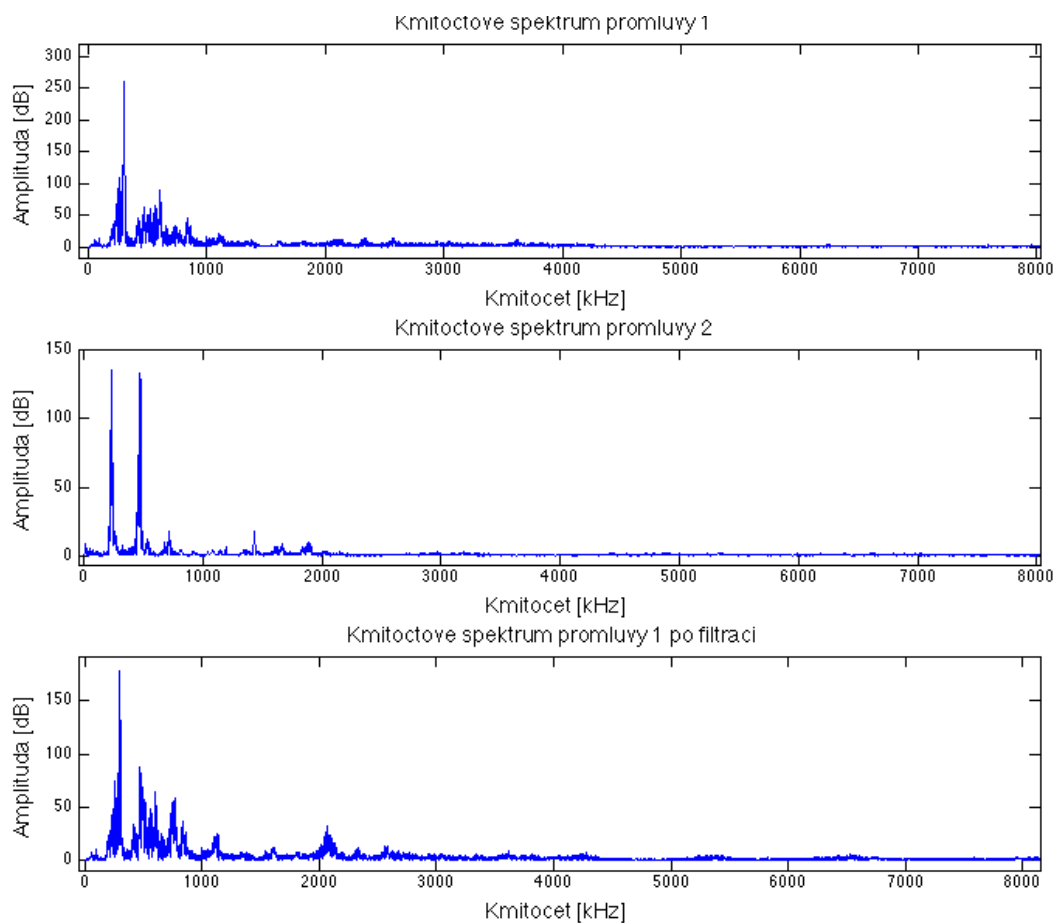


Obr. 5.5: Znázornění filtrů v kaskádě [17]

V případě low- (LF) a high- (HF) shelving filtru f_c je mezní kmitočet a v případě filtrů peak je f_c střední kmitočet, f_b je šířka pásma a G je zisk v dB, $x(n)$ je původní signál a $y(n)$ je výstupní, filtrovaný signál. Promluva je postupně filtrována 13 filtry. První kmitočtové pásmo je filtrováno filtrem typu low-shelving, pak následuje sada 11 peak filtrů a poslední pásmo filtrem typu high-shelving.

V realizovaném programu jsou vypočteny *MFCC* pomocí funkce „mfcc.m” [15]. Dále je podobnost mezi maticemi **MFCC** je vypočtena pomocí funkce „simmx.m” [3] a nejkratší cesta mezi dvěma maticemi **MFCC** pomocí funkce „dp.m” [3]. Matice filtrů je generována funkcí „trifbank.m” [16]. V posledním kroku jsou filtry typů high- a low-shelving a typu peak realizovány funkcemi „shelving.m” [12] resp. „peaking.m” [13].

Kmitočtové spektra promluv lze vidět na obrazu 5.6, kde nahoře je kmitočtové spektrum prvního a pod ní druhého mluvčího. Spodní graf znázorňuje kmitočtové spektrum prvního mluvčího po zesílení kmitočtových pásem podle kmitočtových vlastností mluvčího dva. Na obrazu je vidět, že spodní spektrum není dostatečně změněno, tj. není napodobeno spektrum mluvčího dva. Důvodem může být malé množství filtrů, pomocí kterých nelze přesně změnit spektrum. Výstupní hlas mluvčího se nepodobá výstupnímu hlasu druhého mluvčího. Z tohoto důvodu je nutné tento signál ještě dále zpracovat.



Obr. 5.6: Kmitočtové spektra původních dvou signálů a výstupního signálu

5.2 Změna výšky hlasu

Algoritmus je založen na principu popsaného v části 2.6. Algoritmus postupuje následovně:

1. určení pořadí operací podle α ,
2. změna délky trvání vstupního signálu podle koeficientu posunu,
3. převzorkování.

Na pořadí 2. a 3. kroku nezáleží.

Algoritmus změny délky

Pro změnu délky trvání se používá algoritmus SOLA. Postup algoritmu je následující:

1. segmentace vstupního signálu na segmenty délky N ,
2. segmenty jsou přemístěny podle α ,
3. výpočet korelace mezi přilehlé segmenty,
4. nález bodu kde je hodnota korelace největší, což indikuje bod ve kterém je největší podobnost mezi segmenty,
5. použití indexu tohoto bodu pro určení funkcí fade-in a fade-out a následné aplikace těchto funkcí na segmenty,
6. přidání nových segmentů mezi existující segmenty.

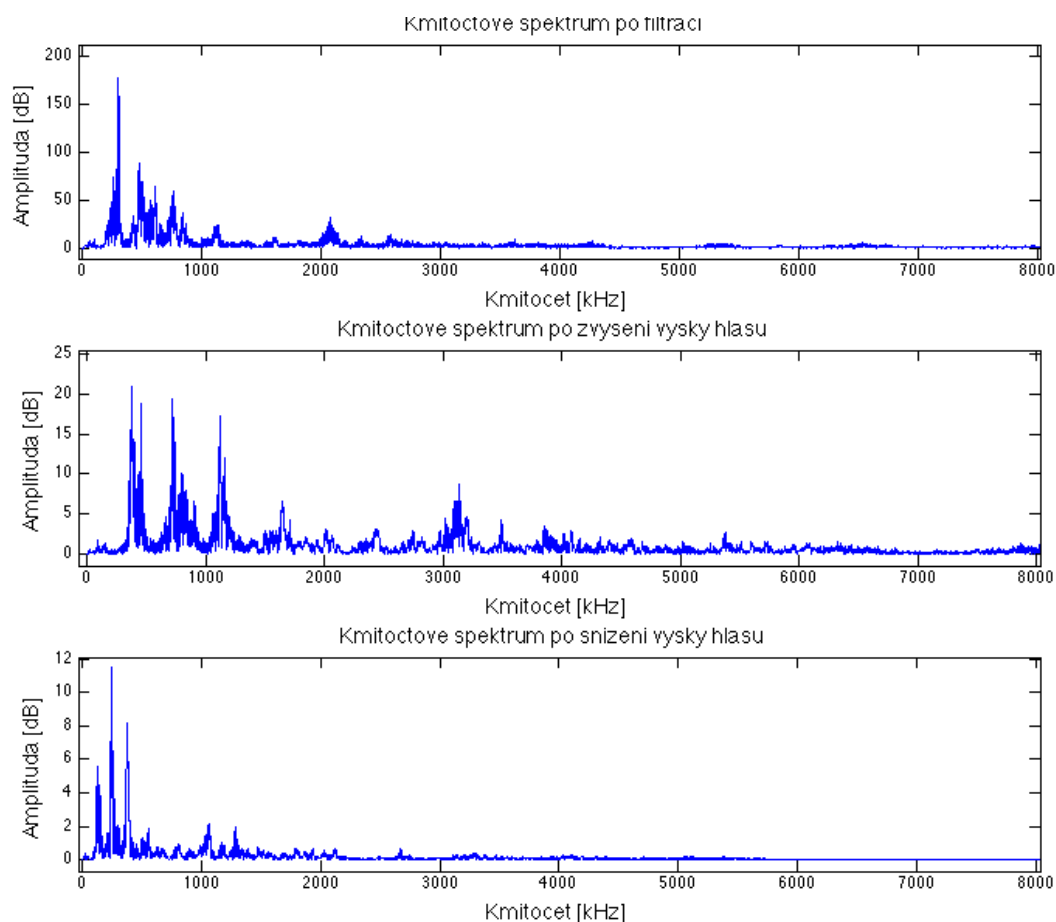
Algoritmus pro převzorkování

Postup algoritmu je následující:

1. pomocí *FFT* je vypočteno frekvenční spektrum vstupního signálu,
2. pro každou pozici kmitočtu v modifikovaném spektru, je určena pozice kmitočtu v původním spektru,
3. hodnota na pozici kmitočtu ve vstupním spektru je kopírována do výstupního spektra,
4. každý nepotřebný kmitočet je vyřazen.

Program byl realizován v Matlabu, kód byl vybrán z knihy Digital Audio Effects [17] a byl modifikován autorem práce [10]. Program umožňuje roztažení v čase s koeficientem roztažení α od 0,25 do 2,00 tzn., je možné signál zúžit až na jeho 25 % a roztáhnout až na dvojnásobek své délky.

Na obrázku 5.7 jsou vidět tři kmitočtové spektra, nahoře je spektrum signálu z předešlé části pod ní jsou dvě spektra, u kterých byla změněna výška hlasu mluvčího. V prvním případě je výška hlasu zvýšena ($\alpha = 1,5$), kmitočtové složky signálu jsou posunuty doprava, tj. směrem k vyšším kmitočtům. Naopak spodní spektrum znázorňuje případ, kdy je výška hlasu snížena ($\alpha = 0,5$), kmitočtové složky signálu jsou posunuty doleva, tj. směrem k nižším kmitočtům. Výstupem tohoto programu je v obou případech de-identifikovaný řečový signál, který je již nerozpoznatelný.



Obr. 5.7: Kmitočtové spektra po změně výšky hlasu

6 EXPERIMENTY

Pro experimenty bylo použito 50 obrazů z databáze *GUFD*. Rozpoznání bylo provedeno pomocí programu „VeriLook”, který ohodnocuje rozpoznání číslem. Tato hodnota vyjadřuje míru správného rozpoznání a je označována jako Skóre. V tabulkách 6.1 a 6.2 je uvedena průměrná hodnota tohoto čísla. Hodnota d vyjadřuje kolik z obrazů bylo rozpoznáno úspěšně.

Tab. 6.1: Výsledky pokusů de-identifikace pomocí základních metod

Metoda	Skóre	d [%]
Žádná	9334	100
Ordinal	357	96
Průměrování	528	92
Bar mask	1037	92
T mask	720	84
Random Noise	86	72
Random Noise Čb	40	38
Negative	23	8
Negative Čb	2	2
Zatemnění	0	0
Mouth Only	0	0
Mouth Only Čb	0	0

Metody v tabulce 6.1 jsou základními metodami de-identifikace obličeje kde nelze nastavit míru de-identifikace. První řádek tabulky je orientační. Na vstupní obrazy nebyla aplikována žádná metoda a tak program rozpoznal úspěšně 100 % obrazů se skórem 9334 což je maximální hodnota. Metoda Ordinal, Průměrování, Bar mask a T mask jsou z pohledu ochrany identity nejméně vyhovující, až 80 % subjektů bylo rozpoznáno úspěšně. Při použití metod Random Noise a Random Noise Černobílý je ochrana osobních údajů nízká, rozpoznalo se úspěšně 72 % resp. 38 % obrazů. Negative a Negative Černobílý jsou metody spolehlivější, avšak nezaručují 100 % ochranu. Metody Zatemnění, Mouth Only a Mouth Only Černobílý úspěšně de-identifikují obličej, žádný subjekt nebyl rozpoznán. Největší skóre mají metody Bar mask, Průměrování, T mask a Ordinal z čehož stejně vyplývá, že jsou nespolehlivé.

Tab. 6.2: Výsledky pokusů základními de-identifikačními metodami při různých hodnotách p , b a t

Metoda	p, b	2	4	6	8	10	12	14	16	18	20
Pixelace	Skóre	4259	1795	668	313	132	79	47	28	34	21
	d [%]	87,5	90	87,5	97,5	77,5	55	40	25	30	20
Rozmazání	Skóre	1797	516	223	126	78	63	46	45	36	28
	d [%]	87,5	90	92,5	82,5	57,5	50	37,5	37,5	32	27
	t	50		100		125		150		175	
Threshold	Skóre	26		139		101		64		19	
	d [%]	18		82		66		52		12	

Metody v tabulce 6.2 jsou základní a zde je možné nastavit míru de-identifikace pomocí koeficientů p , b a t . Při menších hodnotách p a b tj. 2 až 8, jsou metody Pixelace a Rozmazání z pohledu ochrany osobních údajů nespolehlivé, víc než 80 % subjektů bylo úspěšně rozpoznáno postupným zvýšením hodnot p a b počet úspěšných rozpoznání klesá. Z těchto dvou metod je metoda Rozmazání lepší z pohledu de-identifikace, protože skóre této metody jsou menší než při použití Pixelace. Nejhorší výsledky má metoda Threshold při nastavení hodnot t přibližně 100 až 150. Dalším snížením resp. zvýšením této hodnoty lze dosáhnout větší ochrany identity.

Reverzní pokročilá metoda de-identifikace generuje obrazy, které nejsou pro člověka rozpoznatelné. Při ověřování její de-identifikačních vlastností ale byla úspěšnost rozpoznání až 84 %. Z pohledu zachování nejvyšší kvality v de-identifikovaném obrazu je tato metoda nejefektivnější, ale z pohledu ochrany soukromí je tato metoda neefektivní. Důvodem může být skutečnost, že je v obrazu změněna jen textura a nikoliv tvar. Protože *ASM* se během de-identifikace nemění tak i program rozpoznání, který zná tvar původního obličeje, může rozpoznat obličej.

Ověření vlastností de-identifikovaného řečového signálu nebylo možné provést, protože nebyl k dispozici program rozpoznání hlasu.

7 ZÁVĚR

Hlavním cílem této bakalářské práce je seznámení čtenáře s metodami de-identifikace obličejů a metodami de-identifikace hlasu.

V první a druhé kapitole byly popsány jednotlivé metody de-identifikace obličeje resp. hlasu. K realizaci programu a návrhům algoritmů jsou věnovány samostatné kapitoly. Na základě principů první kapitoly byl realizován de-identifikační program, který je popsán v kapitole třetí. Z pohledu ochrany soukromí je nejefektivnější metodou Zatemnění, protože je zde zcela odstraněna veškerá informace o subjektu. Další základní a primitivní metody umožňují, aby se při de-identifikaci zachovalo několik detailů obličeje. Výhodou metod Ordinal, Bar mask a T mask je zachování nejvíce detailů obličeje, ale nejsou efektivními metodami de-identifikace. Metody Pixelace a Rozmazání zachovávají vysokou kvalitu jen při menších hodnotách p a b , při vyšších hodnotách už dojde ke zkreslení, kde neumíme rozeznat, jestli je na obraze obličej či nikoliv. Z pohledu zachování vysoké kvality obrazu je efektivnější metodou k -Same a k -Same-Select. Tyto metody zajišťují vyšší kvalitu i při velkém zkreslení.

V kapitole třetí byl navržen algoritmus reverzní de-identifikace obličeje na základe principů první kapitoly. Podle algoritmu byl realizován program, pomocí kterého lze ovlivnit vizuální vlastnosti výstupních obličejů. Metoda zachovává nejvyšší kvalitu výstupních obličejů, avšak výsledné obrazy obličeje nejsou člověkem rozpoznatelné. Jeho nevýhodou je, že nezaručuje ochranu soukromí proti programu rozpoznání obličeje, jelikož se v obraze mění jen textura a nikoliv tvar obličeje.

V kapitole čtvrté byl navržen algoritmus de-identifikace hlasu pomocí změn hladin energií v daných kmitočtových pásmech. Protože hlas mluvčího není dostatečně změněn, řečový signál je dále zpracován tak, aby výška hlasu mluvčího byla změněna. Výstupem tohoto algoritmu je nerozpoznatelný řečový signál.

V kapitole páté byly uvedeny výsledky experimentů s de-identifikovanými obrazy pomocí realizovaných programů. Byly zde vyhodnocené metody základní, primitivní a pokročilé a také de-identifikační metoda reverzní.

LITERATURA

- [1] BURTON, A. M., D. WHITE a A. MCNEILL. The Glasgow Face Matching Test. *Behavior Research Methods*. 2010, vol. 42, č. 1, s. 286-291. DOI: 10.3758/BRM.42.1.286.
- [2] DESAI, N., K. DHAMELIYA a V. DESAI. Recognizing voice commands for robot using MFCC and DTW. *International Journal of Advanced Research in Computer and Communication Engineering*. 2014, vol. 3, č. 5. ISSN 2278-1021
- [3] ELLIS, D. *Dynamic Time Warp (DTW) in Matlab*. 2003. Dostupné z: <<http://www.ee.columbia.edu/~dpwe/resources/matlab/dtw/>>.
- [4] GROSS, R., E. AIROLDI, B. MALIN a L. SWEENEY. Integrating Utility into Face De-identification. *Privacy Enhancing Technologies*. Springer Berlin Heidelberg, 2006, s. 227-242. DOI: 10.1007/11767831_15.
- [5] GROSS, R., L. SWEENEY, J. COHN, F. DE LA TORRE a S. BAKER. Face De-identification. *Protecting Privacy in Video Surveillance*. London: Springer London, 2009, s. 129. DOI: 10.1007/978-1-84882-301-3_8.
- [6] KEOGH, E. a A. RATANAMAHATANA. Everything you know about Dynamic Time Warping is Wrong. *The third SIGKDD workshop on mining temporal and sequential data (KDD/TDM 2004)*. 2004.
- [7] LI, S. Z. a A. K. JAIN. *Handbook of face recognition*. 2nd ed. London: Springer, 2011. ISBN 978-085-7299-321.
- [8] MATTHEWS, I. a S. BAKER. Active Appearance Models Revisited. *International Journal of Computer Vision*. 2004, vol. 60, issue 2, s. 650-663. DOI: 10.1007/978-3-642-37431-9_50.
- [9] NEWTON, E.M., L. SWEENEY a B. MALIN. Preserving privacy by de-identifying face images: Hiding a Face in a Face. *IEEE Transactions on Knowledge and Data Engineering*. 2005, vol. 17, issue 2, s. 651-661. DOI: 10.1007/978-3-540-76390-1_64.
- [10] RAHMAN, S. *Pitch shifting of voices in real-time*. 2008. ENGR446 Report. University of Victoria.
- [11] STEGMANN, M.B., B.K. ERSBOLL, R. LARSEN. Fame-a flexible appearance modeling environment. *IEEE Transactions on Medical Imaging*. 2003, 22(10), s. 1319-1331. DOI: 10.1049/pbce068e_ch18.

- [12] TACKETT, J. *Bass / Treble Shelving Filter*. 2007. Dostupné z: <http://www.mathworks.com/matlabcentral/fileexchange/16568-bass---treble-shelving-filter/content/shelving.m>.
- [13] TACKETT, J. *Peaking / Notch IIR Filter*. 2007. Dostupné z: <http://www.mathworks.com/matlabcentral/fileexchange/16567-peaking-notch-iir-filter/content/peaking.m>.
- [14] TZIMIROPOULOS G. a M. PANTIC. *Optimization problems for fast AAM fitting in-the-wild*. ICCV 2013
- [15] WOJCICKY, K. *Htk MFCC Matlab*. 2011. Dostupné z: <http://www.mathworks.com/matlabcentral/fileexchange/32849-htk-mfcc-matlab/content/mfcc/mfcc.m>.
- [16] WOJCICKY, K. *Htk MFCC Matlab*. 2011. Dostupné z: <http://www.mathworks.com/matlabcentral/fileexchange/32849-htk-mfcc-matlab/content/mfcc/trifbank.m>.
- [17] ZÖLZER, Udo. *DAFX: digital audio effects*. 2nd ed. Chichester: Wiley, 2011, xxi, 602 s. ISBN 978-0-470-66599-2.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

AAM	aktivní modely vzhledu (<i>Active Appearance Models</i>)
ASM	aktivní modely tvaru (<i>Active Shape Models</i>)
DFT	diskrétní Fourierova transformace (<i>Discrete Fourier Transform</i>)
DP	dynamické programování (<i>Dynamic Programming</i>)
DTW	dynamické borcení času (<i>Dynamic Time Warping</i>)
FBE	energie kmitočtových pásem (<i>Frequency Band Energy</i>)
FFT	rychlá Fourierova transformace (<i>Fast Fourier Transform</i>)
GUFD	obrazová databáze tváří (<i>Glasgow Unfamiliar Face Database</i>)
IFFT	inverzní rychlá Fourierova transformace (<i>Inverse Fast Fourier Transform</i>)
IMM	obrazová databáze tváří (<i>Informatics and Mathematical Modelling</i>)
MFCC	melovské frekvenční keprstrální koeficienty (<i>Mel Frequency Cepstral Coefficients</i>)
SOLA	synchronní překrytí a přidání (<i>Synchronous Overlap and Add</i>)

A PŘÍLOHY

A.1 Obsah přiloženého CD

I Bakalářská práce v elektronické verzi

II De-identifikace obličeje

- Pictures (složka obsahující obrazy)
- Positions (složka obsahující pozice hlavních rysů obličeje)
- KSAME (složka potřebná pro funkci metody k-Same)
- deidOblicej_HPm.m (hlavní program realizovaný pomocí *menu*)
- deidOblicej_HPg.m (hlavní program realizovaný pomocí *GUIDE*)

III Reverzní de-identifikace obličeje

- Funkce (složka obsahující pomocné funkce)
- Pictures (složka obsahující obrazy)
- Positions (složka obsahující pozice hlavních rysů obličeje)
- Proměnné (složka obsahující pomocné proměnné)
- Rev_deid.m (hlavní program)

IV De-identifikace hlasu

- Funkce (složka obsahující pomocné funkce)
- Promluvy (složka obsahující promluvy)
- deidHlas_HP.m (hlavní program)

Programy byly realizovány a testovány v programu Matlab 2014a.