AKADEMIA GÓRNICZO-HUTNICZA

im. Stanisława Staszica w Krakowie

WYDZIAŁ INŻYNIERII MECHANICZNEJ I ROBOTYKI

Magisterska praca dyplomowa

Piotr Chmielowski
Imię i nazwisko

Inżynieria Akustyczna
Kierunek studiów

Rekomendacja utworów muzycznych na podstawie automatycznej analizy emocji w muzyce

Temat pracy dyplomowej

dr inż. Jakub Gałka
Promotor pracy

……..
Ocena, data, podpis Promotora

Kraków, rok 2014/2015

Spis treści

Streszczenie

Praca magisterska poświęcona jest zagadnieniu automatycznego rozpoznawania nastroju utworów muzycznych. Została w niej omówiona motywacja stojąca za rozwijaniem tej dziedziny inżynierii wraz z wynikami przeprowadzonej ankiety na temat przyzwyczajeń słuchaczy oraz szerszymi obserwacjami na temat ewolucji sposobów doboru repertuaru muzycznego na przestrzeni lat. Motywacja podparta została wnioskami i rozważaniami dotyczącymi przyszłości rynku narzędzi do słuchania muzyki.

Następnie został dokonany przegląd literatury naukowej dotyczącej zagadnienia, zaprezentowane zostały także powszechnie używane metody matematyczne i inżynierskie wspierające rozpoznawanie nastroju. Wyciągnięte wnioski przyczyniły się do wyboru narzędzi na których zbudowany został omawiany w tej pracy system.

Główną częścią niniejszej pracy dyplomowej jest opis tworzonego w jej ramach systemu rozpoznającego nastrój utworów muzycznych. Celem zadania było zbudowanie oprogramowania, które byłoby atrakcyjne i użyteczne dla słuchaczy. W trakcie jego rozwoju, były wyciągane wnioski na temat skuteczności używanych metod. Celem poznawczym pracy było odpowiedzenie na pytanie jaki wpływ na skuteczność rozpoznawania nastroju ma dobór cech obliczonych na podstawie reprezentującego nagrania sygnału audio: współczynników mel-cepstralnych oraz przesuwanych delta współczynników.

Weryfikacja działania systemu pozwoliła na wyciągnięcie opisanych w pracy wniosków oraz sformułowanie zaleceń, które mogą okazać się pomocne przy dalszym rozwijaniu systemu a także innych prac z omawianej dziedziny. Skomentowano zaobserwowane błędy w działaniu klasyfikatora, dokonany analizy możliwych przyczyn ich powstania oraz sformułowano zestaw możliwych ich rozwiązań.

Rozdział 1

Wstęp

1.1 Motywacja

Niniejsza praca magisterska została zainspirowana następującym pytaniem: ***Jak w przyszłości będzie wyglądało słuchanie muzyki?*** a dotyczy ono tego jak pomagać słuchaczom będzie technologia.

1.1.1 Jak słuchamy muzyki dzisiaj?

W XX wieku słuchano muzyki z mało wygodnych nośników. Dostęp do konkretnych utworów wymagał zmiany płyty czy przewinięcia kasety, słuchano więc najczęściej całych albumów. Źródłami rekomendacji byli znajomi, stacje radiowe i gazety muzyczne, a poleconej muzyki można było słuchać dopiero po zakupie czy przegraniu nośnika. Repertuar był zaś ograniczony wielkością posiadanej kolekcji płyt czy kaset.

Dzisiaj dostęp do muzyki jest błyskawiczny: zarówno dzięki możliwości taniego i szybkiego kupowania pojedynczych utworów (iTunes, Muzodajnia), jak i dzięki odtwarzaniu strumieniowemu (Spotify, Youtube). W efekcie repertuar dobiera się - często na bieżąco, podczas słuchania - z kompozycji pochodzących z różnych albumów.

1.1.2 Jak będziemy robić to jutro?

W ciągu ostatnich 30 lat rozwoju elektroniki użytkowej, największy sukces osiągały produkty, które zapewniały użytkownikowi prostotę obsługi. Przykładem jest zwycięstwo systemów operacyjnych z graficznym interfejsem użytkownika nad tymi z tekstowym. Użytkownicy oczekują też, że włączenie danej funkcji wymagać będzie jak najmniejszej liczby akcji (szczególnie ważne gdy urządzenie jest używane podczas uprawiania sportu czy prowadzenia pojazdów). Rozwój amatorskich aparatów cyfrowych jest z kolei przykładem tendencji do ograniczania liczby przycisków.

Funkcja tasowania utworów w odtwarzaczach muzycznych

Trend upraszczania i ułatwiania obsługi dotyczy także urządzeń i oprogramowania do słuchania muzyki. W większości odtwarzaczy znajduje się tzw. tryb *shuffle* - losowe odtwarzanie nagrań. Istnieją nawet odtwarzacze pozbawione wyświetlacza, w których główną metodą doboru repertuaru jest właśnie ta funkcja (np. Apple iPod Shuffle).

Funkcjonalność ta wiąże się jednak z pewnymi niedogodnościami - podczas gdy popularność zdobywają urządzenia, które można dostosować do aktualnych potrzeb w prosty i szybki sposób - tryb tasowania losuje utwory z całej kolekcji użytkownika lub ręcznie (czyli w czasochłonny sposób) wybranego jej podzbioru.

Idąc jeszcze dalej, warto zauważyć, że coraz popularniejsza staje się komunikacja między użytkownikiem a urządzeniem oparta na odgadywaniu jego intencji - na przykład programy typu Siri czy WolframAlfa czy personalizacja wyszukiwań w Google. Na tym polu tryb tasowania również przegrywa - oferuje jedynie zwykłe losowanie.

Potencjalnym problemem więc może być to, że - podczas gdy kolejność utworów na albumach jest przez muzyków i producentów tak dobierana, aby dawała efekt estetyczny - w przypadku słuchania losowego, często zdarza się sytuacja, że następujące po sobie kompozycje do siebie całkowicie nie pasują, ich następstwo jest wręcz nieprzyjemne dla słuchacza.

Badania jakościowe dotyczące zwyczajów słuchania muzyki

Aby znaleźć potwierdzenie lub zaprzeczenie dla wniosków postawionych w poprzedniej sekcji, przeprowadzono badania na temat preferencji słuchaczy co do sposobu wyboru repertuaru do słuchania - głównie używania funkcji tasowania i ewentualnych niedogodności z nią związanych. Wzięło w nich udział około 20 osób - głównie znajomi autora ale także użytkownicy for internetowych.

Respondentom zostały zadane następujące pytania:

1. Czy często używasz trybu losowego odtwarzania?
2. Kiedy go używasz, pozwalasz programowi losowo wybierać piosenki z całej kolekcji, czy wcześniej wybierasz z niej zbiór płyt, wykonawców lub utworów? Nie przeszkadza Ci przeznaczanie czasu na dokonywanie tego wyboru, czy najchętniej zrobiłbyś to jak najszybciej?
3. Kiedy używasz tego trybu, czy często omijasz poszczególne utwory? Jeśli omijasz - dlaczego? Czy często następuje to, jeśli po kolejna piosenka bardzo się różni nastrojem lub energicznością od poprzedniej?

Wszyscy respondenci stwierdzili, że używają funkcji tasowania utworów
Głównym problemem sygnalizowanym przez badanych jest niedopasowanie nastroju kolejnych utworów: np. bezpośrednio po energicznej, wesołej kompozycji odtwarzana jest spokojna i melancholijna. Słuchacze wolą kiedy zmiany

nastrojów następują bardzo powoli - na przestrzeni wielu utworów. Badani dążą też do tego aby jak najmniej czasu poświęcać na wybór muzyki do słuchania oraz aby ten proces był jak najłatwiejszy.

Biorąc pod uwagę, że było to badanie jakościowe a nie ilościowe, oraz zostało przeprowadzone na niewielkiej grupie osób, nie ma możliwości wyciągnięcia wniosków opisujących zachowanie całego społeczeństwa. Można jednak spodziewać się, że wyniki te oznaczają, że istnieje pewna grupa odbiorców wśród których obserwacje z poprzednich sekcji są aktualne.

Oznacza to, że przed autorami oprogramowania do słuchania muzyki stoi wyzwanie - powinni oni dostarczyć słuchaczom narzędzia, które zastąpią producentów płyt, którzy starannie dobierali kolejność piosenek. Kluczem zaś według którego mają być dobierane utwory do jednej listy odtwarzania, powinien być nastrój.

1.2 Technologia wychodząca na przeciw trendom w słuchaniu muzyki

Jednym z możliwych podejść do ulepszenia trybu tasowania, byłoby ręczne sklasyfikowanie dostępnej muzyki pod kątem nastroju (takie próby są prowadzone np. Allmusic.com) a następnie wyposażenie trybu tasowania w funkcją dobierania utworów z specyficznym nastrojem. Zadanie to jest jednak bardzo pracochłonne i nie ma możliwości aby jego zasięg był szeroki (sklasyfikowanie całej stworzonej przez człowieka muzyki byłoby niemal niemożliwe, a z pewnością bardzo kosztowne).

Pozostaje więc stworzenie takiego systemu, który sam określa nastrój muzyki. Najlepszym rozwiązaniem jest stworzenie aplikacji, która zostanie wytrenowana na zbiorze utworów, których nastrój jest znany, a następnie będzie potrafiła uogólnić tą wiedzę i sklasyfikować według nastroju nowe, nie znane sobie wcześniej utwory.

Obecne na rynku systemy doboru repertuaru według nastroju

Istnieje już kilka aplikacji, które działają właśnie w taki sposób. Niektóre z nich operują na kolekcji utworów użytkownika, niektóre połączone są z własnym serwisem strumieniowego odtwarzania muzyki.

Poniżej znajduje się ich lista najpopularniejszych:

Musicovery Aplikacja z *front-endem* w przeglądarce internetowej oraz dla urządzeń z systemem Android i iOS. Oferuje odtwarzanie strumieniowe z własnej kolekcji muzyki oraz kolekcji użytkownika (tylko wersja mobilna). Użytkownik wybiera nastrój na płaszczyźnie, której osiami współrzędnych są: wartościowość (smutne - wesołe) i pobudzenie (spokojne - energiczne).

Aupeo Oferuje tylko odtwarzanie strumieniowe, *front-end* dla przeglądarki internetowej i systemów Android, iOS oraz Windows Phone. Można wybrać jeden z 10 nastrojów, których lista znajduje się w aplikacji.

Stereomood Odtwarzanie strumieniowe dla przeglądarki, Androida i iOS. Użytkownik wybiera z listy kilkudziesięciu nastrojów oraz okoliczności.

Moodagent Serwis istnieje w formie wtyczki dla serwisu Spotify oraz niezależnej aplikacji dla systemu Android i Windows Phone. Nastrój wybierany jest poprzez zestaw pięciu suwaków reprezentujących różne składowe emocji. Jest także możliwość zbudowania listy odtwarzania w oparciu o nastrój wybranego utworu.

Większość z nich nie pozwala na użycie w procesie wyboru utworów z własnej kolekcji. Wielu użytkowników jest jednak przyzwyczajonych do słuchania nagrań zapisanych na ich dysku, zamiast odtwarzania strumieniowego.

1.3 Udział niniejszej pracy w rozwoju narzędzi do słuchania muzyki

1.3.1 Cele pracy

Niniejsza praca magisterska ma dwa cele - pierwszy z nich to znalezienie (w tym: porównanie różnych), zaimplementowanie i przetestowanie jak najlepszej metody rozpoznawania nastroju, a także odpowiedź na postawione w kolejnej sekcji pytania poznawcze.

Drugi to wykorzystanie tej metody do stworzenia prototypu systemu, który będzie użyteczny dla użytkownika, który może stanowić konkurencję dla oprogramowania wymienionego w sekcji 1.2. Częścią tego systemu dostępną bezpośrednio dla użytkownika ma być wtyczka (rozszerzenie) do jednego z popularniejszych odtwarzaczy muzycznych - programu Winamp.

Powodem wyboru tego odtwarzacza jest to, że jest on dość popularny, ponadto pozwala w łatwy sposób tworzyć i integrować rozszerzenia i nowe funkcje. Główna część systemu (klasyfikująca utworu) nie jest jednak z nim powiązana co pozwala w przyszłości zaimplementować wtyczki dla innych odtwarzaczy.

Hipotezy i pytania dotyczące metod parametryzacji i klasyfikacji sygnału

- Postawiona została hipoteza, że uwzględnienie cech SDC wpłynie na poprawę jakości działania klasyfikatora.
- Postawiona została też hipoteza, że użycie algorytmu analizy głównych składowych i redukcja długości superwektora przed trenowaniem i testowaniem klasyfikatora wpłynie pozytywnie na skuteczność klasyfikacji.

- Jakie są optymalne parametry poszczególnych metod statystycznych i klasyfikatorów:
  liczby wymiarów po redukcji superwektora algorytmem PCA,
  współczynników C i γ maszyny wektorów wspierających. [1]

1.3.2 Układ pracy

W rozdziale 2 dokonany jest przegląd literatury naukowej związanej z zagadnieniem rozpoznawania nastroju, a także omówione są wnioski z tego przeglądu. Znajduje się tam również opis tego jakie decyzje co do użytych w implementowanym systemie metod zostały podjęte, a same metody są opisane z teoretycznego oraz praktycznego punktu widzenia.

W rozdziale 3 udokumentowane są poszczególne komponenty wchodzące w skład systemu informatycznego służącego do osiągnięcia celów pracy.

Rozdział 4 opisane są procedury w jakich zostały użyte wspomniane wyżej komponenty w celu przygotowania systemu do jego weryfikacji oraz pod potrzeby użytkownika, a także procesy wcześniejszego przygotowania danych.

Rozdział 5 zawiera omówienie otrzymanych wyników klasyfikacji. Znajduje się w nim także komentarz dotyczący odpowiedzi na postawione w sekcji 1.3.1 pytania poznawcze dotyczące wpływu zastosowanych metod parametryzacji sygnału na skuteczność klasyfikacji nastrojów. Zaprezentowane są również możliwe scenariusze dalszego rozwoju i wykorzystania systemu oraz wykorzystanych w niniejszej pracy metod.

Rozdział 6 zawiera podsumowanie wykonanej pracy oraz opisuje w jakim stopniu osiągnięte zostały wyznaczone cele.

[1] Wspomniane skróty i metody zostaną wyjaśnione i omówione w rozdziale 2

Rozdział 2

Część teoretyczna

2.1 Przegląd literatury

Dział nauki zajmujący się automatycznym pozyskiwaniem informacji z muzyki nazywa się w literaturze angielskojęzycznej **Music Information Retrieval**, w skrócie **MIR**.

Łączy on zagadnienia z zakresu przetwarzania sygnałów cyfrowych, muzykologii jak i sztucznej inteligencji czy uczenia maszynowego, a nawet psychologii. Zajmuje się on takimi sprawami jak rozpoznawanie utworów muzycznych, gatunków muzyki, wykonawców a także nastroju.

Od kilkunastu lat naukowcy z dziedziny MIR zajmują się poszukiwaniem rozwiązania problemu automatycznego wykrywania nastroju. Od roku 2007 w ramach corocznych konferencji MIREX (The Music Information Retrieval Evaluation eXchange) organizowany jest konkurs na najbardziej skuteczny system rozpoznawania nastroju [17].

Uwaga: W literaturze proces przypisywania utworom (w szczególności plikom dźwiękowym) metadanych (takich jak nazwisko autora, rok nagrania, nastrój itp) określany jest terminem *tagowanie*. W niniejszej pracy poprzez tagowanie rozumiany jest proces przypisywania plikowi nazwy nastroju.

Generalnym podejściem do tematu rozpoznawania nastroju jest:

1. zebranie dużej bazy nagrań, których nastrój jest znany
2. przekształcenie każdego nagrania (wektora bardzo wielu próbek) w wektor o niższym wymiarze (najczęściej w procesie parametryzacji, po której następuje uśrednienie niektórych cech, czasem eliminacja najmniej znaczących lub redundantnych parametrów)
3. wytrenowanie klasyfikatora w oparciu o wspomniane wektory oraz odpowiadające im klasy nastrojów

Do rozpoznania nastroju nowych utworów używa się tego samego klasyfikatora i identycznej procedury sporządzania wektora cech.

Cechy dzielą się na te związane z całym utworem (lub kilkusekundowym fragmentem) takie jak tempo czy tonacja oraz te związane z ramkami (bardzo krótkimi fragmentami nagrania - najczęściej 20-50 ms). Cechy związane z ramkami są często reprezentowane przez ich parametry statystyczne.

Baza danych treningowych

Laurier [19] w swojej rozprawie doktorskiej zaprezentował podsumowanie części prac naukowych dotyczących omawianej dziedziny. Jednym z płynących z niej wniosków jest fakt, że liczba utworów do treningu klasyfikatora waha się między 110 a 1240 w zależności od autorów rozwiązania.

Jest kilka sposobów na pozyskiwanie informacji o tym w jaki sposób określać nastrój danego:

- tagowanie przez jednego słuchacza
- tagowanie przez grupę ekspertów (np.: [28] lub baza Allmusic.com)
- *crowdsourcing* - tagowanie przez dużą liczbę słuchaczy/użytkowników serwisu np. przy pomocy interaktywnej gry MoodSwings opisywanej przez Kima i innych w [18] lub z bazy Last.fm

Model nastrojów

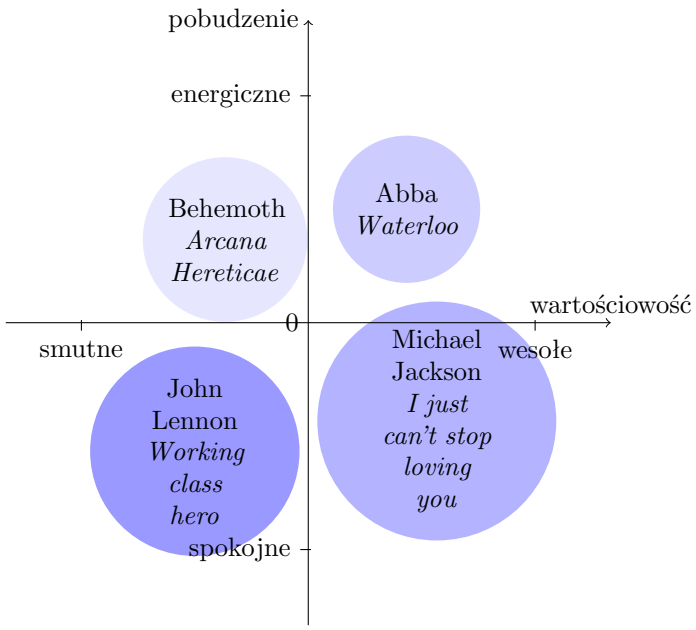
W przeprowadzonych wcześniej pracach na omawiany temat można wyróżnić dwa sposoby przypisywania utworom nastrojów:

- klasyfikacja piosenek do kategorii z nazwami nastroju
- mapowanie (regresja) na ciągłej przestrzeni nastrojów - najczęściej dwuwymiarowej - płaszczyzny, gdzie jedna oś to **wartościowość** (parametr określający czy utwór jest smutny czy wesoły), druga: **pobudzenie** (parametr określający czy utwór jest energiczny czy spokojny). Płaszczyznę taką przestawia rysunek 2.1.

Portal Allmusic.com używa 179 kategorii do opisania nastroju, przy czym Hu i Downie [16] zauważyli, że pewna ich część jest przypisana do niewielkiej liczby utworów, między pozostałymi zaś występuje dość duża korelacja (przykładowo: *Fun* - zabawa, rozrywka i *Cheerful* - pogodny, wesoły).

Wybrali więc 40 tagów, przeprowadzili analizę podobieństwa i w jej wyniku utworzyli 5 klastrów (grup), z których każdy zawiera 5-7 tagów, przy czym żaden tag nie należy do więcej niż jednego klastra. Takie podejście (5 klastrów) jest stosowane w konkursie MIREX [1].

W bazie Allmusic.com, tagi dotyczące nastrojów są przypisywane do utworów arbitralnie przez pracowników portalu i *"opisują brzmienie oraz odczucia dotyczące utworów"* [2].



Rysunek 2.1: Płaszczyzna pobudzenie-wartościowość z zaznaczonymi przykładowymi nagraniami

Parametryzacja i modelowanie statystyczne

Zdecydowana większość autorów przyjmuje, że wystarczy aby system rozpoznający nastrój brał pod uwagę tylko fragment utworu muzycznego (zarówno w procesie trenowania jak i klasyfikacji).

Konkurs MIREX wymaga, aby wysyłane programy wczytywały 30-sto sekundowe fragmenty audio [1]. Zaletą takiego rozwiązania nad przetwarzaniem całego nagrania jest większa wydajność systemu (mniej danych do przetworzenia skutkuje mniejszym czasem i mocą obliczeniową wymaganą do przeprowadzenia procesu).

Li i Ogihara [20] oraz Laurier [19] używali 30-sekundowych fragmentów. Panagakis i Kotropoulos stosowali fragmenty o długości 10-30 sekund [23]. Z kolei Pampalk i inni w swojej pracy przyjęli, że do rozpoznania charakteru piosenki przez człowieka, wystarczy fragment o długości 6 sekund [22].

Należy zauważyć, że w takim przypadku przyjmuje się założenie, że nastrój fragmentu jest reprezentatywny dla całej piosenki, a co za tym idzie - nastrój całego utworu jest stały. Oczywiście jest to uproszczenie i w przypadku wielu utworów jest ono nieprawdziwe, dlatego w przyszłości należy zastanowić się nad rozwiązaniem tego problemu w inny sposób.

Poniżej opisano przykładowe (najbardziej popularne) podejścia do tematu parametryzacji i modelowania statystycznego stosowane przez autorów w ostatnich

latach.

Bergstra i inni w swojej pracy [7] dzielą utwór na niewielkie ramki (między 25 a 50 ms), dla każdej z nich obliczają współczynniki mel-cepstralne (opis w 2.3.3), a następnie łączą je w segmenty (tzw. okna teksturalne) o długości około 4 sekund (80 ramek), reprezentowane przez policzone dla całego segmentu średnie dla każdego z 32 współczynnika oraz ich macierz kowariancji. Podobnie robią Seyerlehner i inni [26].

Podobnie Burred i inni [10] dzielą utwór na ramki o długości 60 ms (zachodzące na siebie w 20 ms) i obliczają 280 różnych cech dla każdej z nich. W kolejnym kroku obliczają średnią i odchylenie standardowe każdej cechy dla całego sygnału. W stosunku do poprzedniego rozwiązania, autorzy nie liczą zwykłej średniej arytmetycznej, ale średnią ważoną głośnością ramki.

Grecu i Rauber [15] dzielą sygnał na ramki o długości 50 ms, obliczają 24 zmodyfikowane współczynniki mel-cepstralne. W kolejnym kroku biorą, dla każdego współczynnika, jego 32 kolejne wartości i obliczają transformatę Fouriera. Otrzymują w ten sposób segmenty (około 12 sekundowe) z 768 wartościami. Każda z tych wartości jest uśredniana po wszystkich segmentach pliku audio.

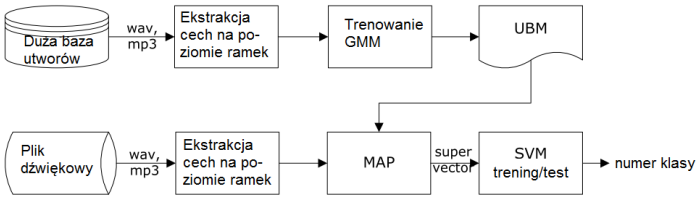
Mandel i Ellis [21] obliczają między innymi współczynniki mel-cepstralne dla ramek o długości 25 ms, a następnie tworzą wektor dla 10-sekundowego fragmentu pliku dźwiękowego poprzez policzenie średniej oraz macierzy kowariancji (ze zlogarytmowanymi współczynnikami na diagonali, rozwiniętej w wektor) dla współczynników zawartych w tym fragmencie. Następnie uśredniają wyniki pomiędzy 5 kolejnych (nachodzących na siebie) segmentów.

Burred i Lerch w [9] dzielą sygnał na ramki, dla każdej tworzą wektor 20 cech takich jak współczynniki mel-cepstralne, środek ciężkości widma itp, a następnie dla całego fragmentu utworu tworzą 80-cio elementowy wektor składający się z średnich i odchyleń standardowych dla każdej cechy oraz, dodatkowo, jej pochodnej. Dodają do tego 10 innych cech związanych z rytmem oraz liczba ramek o niskiej energii. Przeprowadzają następnie test odporności na szum oraz filtrowanie pasmowe dla każdego parametru. W jego wyniku odrzucają 32 cechy.

De Leon i Martinez [14] modelują cechy z ramek przy użyciu średniej i wariancji (używają też cech globalnych związanych z rytmem i tempem).

Byun i inni zaproponowali [11] dodanie do współczynników mel-cepstralne, nowych cech związanych z widmem utworu. Dla każdej z tych cech obliczają średnią, wariancję, a także wartość maksymalną i minimalną w oknie teksturowym o długości 3 sekund.

Poniżej zostaną opisane prace, w których autorzy w procesie obliczania wektora cech korzystali z bardziej zaawansowanych metod modelowania statystycznego.



Rysunek 2.2: Schemat GMM - SVM[12]

Cao i Li w swojej pracy [12] zaimplementowali metodę polegającą na zbudowaniu uniwersalnego modelu tła (opis w 2.3.4) będącego modelem mieszanek gaussowskich na podstawie dużej liczby nagrań. Następnie użyli adaptacji MAP cech kolejnych nagrań do zbudowania tzw. superwektorów oraz maszyny wektorów wspierających (opis w 2.3.2) jako klasyfikatora (Rys. 2.2). Podobne podejście w nowszej pracy zastosował Wu [27].

Innym przykładem jest praca Charbuilleta [13] i innych (używają tylko współczynników mel-cepstralnych i płaskości widma). Do modelowania uniwersalnego modelu tła używają modelu mieszanek gaussowskich z diagonalną macierzą kowariancji. To uproszczenie rekompensują zwiększeniem liczby komponentów w miksturze. Model tła jest trenowany przy użyciu algorytmu Expectation-Maximization (opis w 2.3.4).

W tekście [8] stwierdzone jest, że w przypadku zagadnienia weryfikacji mówcy przy użyciu modelowania mieszanek gaussowskich używa się tylko diagonalnych współczynników macierzy kowariancji, gdyż wg obserwacji, takie rozwiązanie pozwala na uzyskanie większej skuteczności systemu.

Ważną obserwacją jest to, że jakość pliku MP3 ma dość niewielki wpływ na jakość działania systemu rozpoznawania mowy [24]. Jako, że w wspomnianej pracy są używane podobne parametry sygnału jak w niniejszej (między innymi MFCC), uwaga ta staje się użyteczna przy rozpoznawaniu nastroju. Jest o tyle istotna, że słuchacze posiadają utwory w różnej jakości, a celem omawianego systemu jest bycie uniwersalnym.

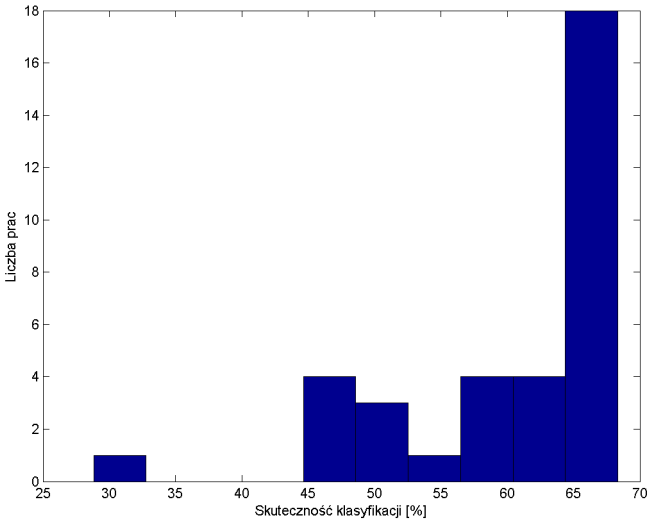
Klasyfikacja

Do mapowania wektorów cech na klasy (nastroje), niemal wszyscy autorzy używają maszyny wektorów wspierających. Stosują oni zestaw takich klasyfikatorów - dla każdego nastroju jeden klasyfikator, który odpowiada na pytanie, czy dane nagranie należy do tej klasy czy nie.

Wyniki klasyfikacji Tabela 2.1 przedstawia analizę statystyczną wyników skuteczności klasyfikacji przeprowadzonej przez algorytmy przesłane na edycję konkursu MIREX w 2013 i 2014 roku, rysunek 2.3 zaś histogram pokazujący rozkład otrzymanych skuteczności klasyfikacji.

Tabela 2.1: Analiza wyników konkursu MIREX

minimum	maksimum	średnia	mediana
28,83%	68,33%	60.04%	65,17%



Rysunek 2.3: Rozkład skuteczność klasyfikacji pośród prac przysłanych na konkurs MIREX w latach 2013-2014

2.2 Cele poznawcze pracy

Po zapoznaniu się z literatura, została wybrana metoda polegająca na:

1. obliczeniu uniwersalnego modelu tła na podstawie danych treningowych
2. dla każdego utworu ze zbioru treningowego obliczenie superwektora (adaptując jego cechy do modelu tła)
3. wytrenowanie klasyfikatora - maszyny wektorów wspierających

Jako cechy, na podstawie których będzie tworzony model tła, wybrane zostały:

- współczynniki mel-ceptralne
- współczynniki SDC (opis w 2.3.3) - zostały wybrane ze względu na szerokie zastosowanie w dziedzinie rozpoznawania języka w nagraniach mowy - temacie, który używa wielu podobnych metod do tych stosowanych przy rozpoznawaniu nastroju w muzyce

Opis nastrojów

W niniejszej pracy nastroje utworów należą do jednej z 4 kategorii zdefiniowanych jako ćwierćpłaszczyzny na płaszczyźnie pobudzenie-wartościowość:

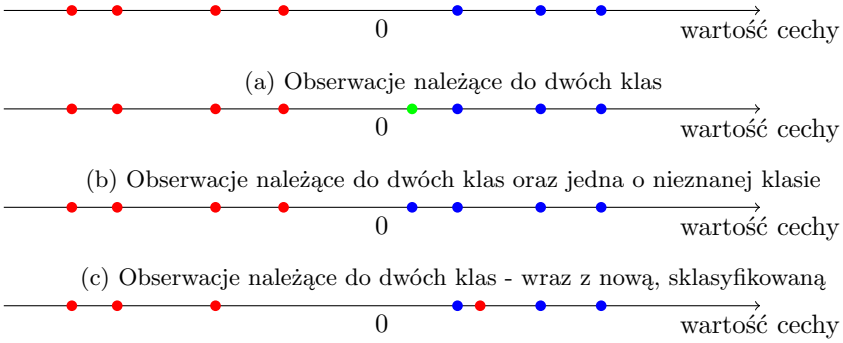
- **klasa 1** - utwory smutny i spokojne
  pobudzenie niskie, wartościowość niska
- **klasa 2** - utwory wesołe i energiczne
  pobudzenie wysokie, wartościowość wysoka
- **klasa 3** - utwory wesołe, pogodne ale spokojne
  pobudzenie niskie, wartościowość wysoka
- **klasa 4** - utwory złowrogie, nerwowe i energiczne
  pobudzenie wysokie, wartościowość niska

Wybór 4 kategorii (zamiast często używanych przez autorów pięciu) został dokonany z dwóch powodów. Po pierwsze należy mieć na uwadze, że ocena nastroju nagrania jest subiektywna i im klas jest więcej, a - co za tym idzie - dokładniej są zdefiniowane, tym większa będzie niezgodność wśród użytkowników w kwestii przypisania konkretnego utworu do jednej z nich. Po drugie mniejsza liczba klas pozwala przyspieszyć wybór nastroju muzyki, której użytkownik zamierza słuchać.

2.3 Opis metod i algorytmów stosowanych przy rozpoznawaniu nastroju

2.3.1 Opis procesu przygotowania, parametryzacji i klasyfikacji utworów muzycznych

Nagranie muzyczna przechowywane jest w pamięci komputera jako sygnał, tj wektor kolejnych próbek. Każda próbka jest liczbą rzeczywistą reprezentującą



(a) Obserwacje należące do dwóch klas

(b) Obserwacje należące do dwóch klas oraz jedna o nieznanej klasie

(c) Obserwacje należące do dwóch klas - wraz z nową, sklasyfikowaną

(d) Obserwacje należące do dwóch klas, których cechy nie są separowalne

fizyczną wielkość - odchylenie membrany głośnika podczas odtwarzania - a co za tym idzie - także ciśnienie akustyczne wywołujące u słuchacza percepcję dźwięku w konkretnej chwili w czasie.

Wszelkie przekształcenia takie jak parametryzacja sygnału polegają na obliczeniach matematycznych stosowanych do wektora próbek.

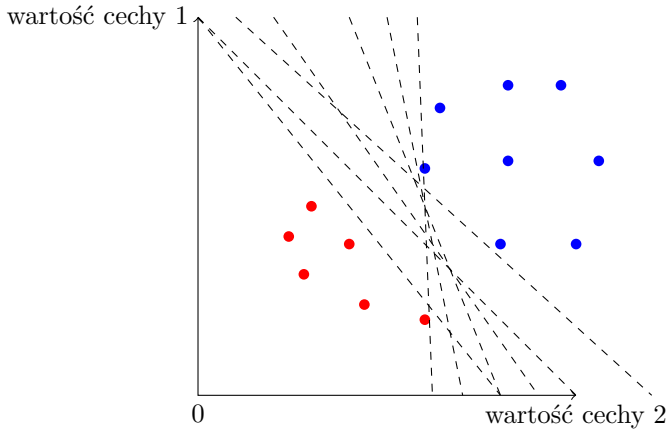
2.3.2 Klasyfikacja statystyczna

Klasyfikacja statystyczna - algorytm, który przydziela obserwacje statystyczne do klas, bazując na ich cechach. Obserwacjami mogą być na przykład fragmenty nagrań audio, a cechami - pewne liczby związane z nimi - na przykład liczba przejść przez zero sygnału dźwiękowego czy jego energia. Klasyfikacja polega wtedy na znalezieniu takiej funkcji, której wejścia to cechy i która zwraca numer klasy, do której ten fragment należy. Klasy są określane arbitralnie - nagrania dźwiękowe mogą należeć na przykład do różnych klas, z których każda reprezentuje określony nastrój.

Algorytm klasyfikujący nazywany jest *klasyfikatorem*. Aby go wyznaczyć, przeprowadza się procedurę trenowania, podczas której znajdowany jest klasyfikator, który prawidłowo funkcjonuje dla danych treningowych: zbioru obserwacji o znanych zarówno cechach jak i klasie, do której należą.

Rysunek 2.4a prezentuje przykład znajdowania klasyfikatora dla prostego przypadku, w którym z każdą obserwacją związana jest jedna cecha. Obserwacje należące do jednej klasy oznaczone są kolorem czerwonym, do drugiej - niebieskim. W tym przypadku jednym z poprawnych klasyfikatorów jest taki, który przypisuje obiekt do klasy pierwszej jeśli wartość cechy jest mniejsza od 0, do drugiej - jeśli jest większa lub równa.

Użycie tego klasyfikatora pokazane jest na rysunku 2.4b. Nowa obserwacja, o nieznanej wcześniej klasie posiada cechę, której wartość jest większa od zera. Można ją więc sklasyfikować jako należącą do klasy drugiej, co pokazuje rysunek 2.4c.



Rysunek 2.5: Obserwacje z dwóch klas, możliwości podziału przestrzeni

Uwaga! Bardzo często nie jest możliwe uzyskanie 100-procentowej skuteczności działania klasyfikatora. Rysunek 2.4d pokazuje przypadek, w którym jeden z obiektów klasy drugiej ma mniejszą wartość cechy niż inny obiekt klasy pierwszej. W tym przypadku należy wyznaczyć taki klasyfikator, który zapewni największą skuteczność klasyfikacji mierzoną współczynnikiem poprawnie zaklasyfikowanych obserwacji do liczby wszystkich.

Najpopularniejsze algorytmy klasyfikacji to maszyna wektorów wspierających, sieci neuronowe itd.

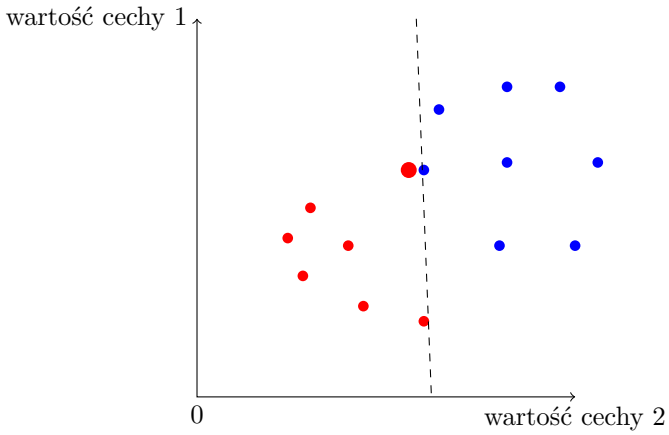
Maszyna wektorów wspierających

Maszyna wektorów nośnych, SVM (z ang. *Support Vector Machine*) – klasyfikator, którego trenowanie prowadzi do znalezienia hiperpłaszczyzny rozdzielającej obserwacje należące do dwóch klas. Najważniejszą cechą tej hiperpłaszczyzny jest to, że rozdziela obserwacje z maksymalnym marginesem - odległością pomiędzy płaszczyzną a najbliższymi jej próbkami.

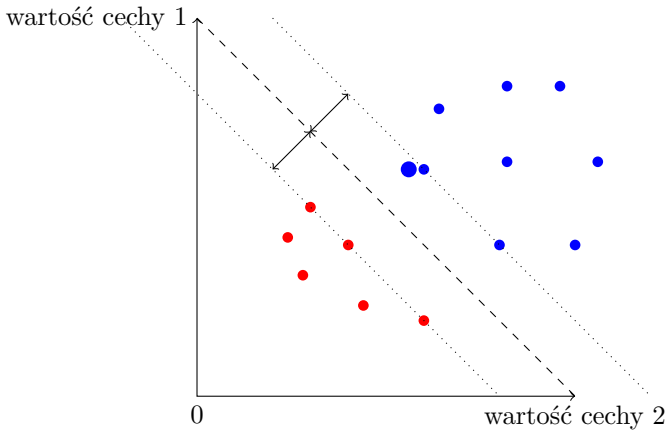
Rysunek 2.5 prezentuje zbiór danych należących do dwóch klas w przestrzeni dwuwymiarowej. W tym przypadku hiperpłaszczyzną jest prosta (jako, że dzieli przestrzeń na dwie części, hiperpłaszczyzna ta ma zawsze wymiar o jeden mniejszy niż wymiar przestrzeni). Jak widać, dane te można podzielić nieskończoną liczbą hiperpłaszczyzn.

Rysunek 2.6 pokazuje przykładową prostą dzielącą próbki treningowe. Jak widać, jej fragmenty znajdują się w bardzo niedalekiej odległości od niektórych próbek. Prowadzi to do sytuacji, w której nieznana podczas treningu próbka (zaznaczona na rysunku jako większe koło) zostanie sklasyfikowana jako należąca do klasy A, podczas gdy jej cechy są bliższe cechom próbek z klasy B.

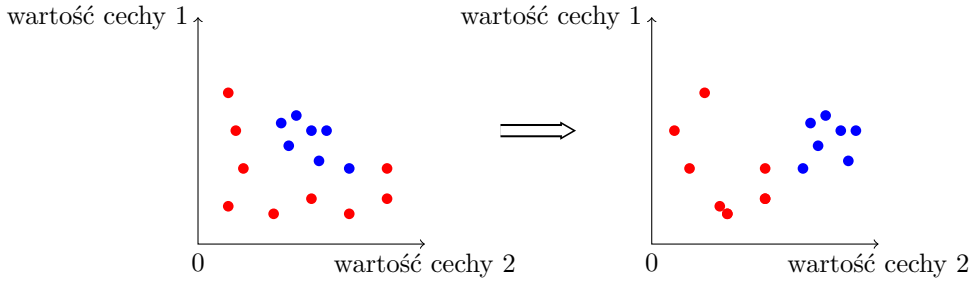
Aby temu zapobiec, poszukiwana jest taka hiperpłaszczyzna, która znajduje się w jak najdalszej odległości od najbliższych jej próbek obu klas. Na rysunku 2.7 została zaznaczona właśnie taka prosta. Podkreślone też są jej odległości od



Rysunek 2.6: Obserwacje z dwóch klas, nieoptymalny podział przestrzeni



Rysunek 2.7: Obserwacje z dwóch klas, optymalny podział przestrzeni



Rysunek 2.8: Przekształcenie przestrzeni do takiej, w której klasy są liniowo separowalne

najbliższych jej próbek (marginesy). Zaznaczona jest także nowa - ta sama co na poprzednim rysunku - obserwacja - tym razem zaklasyfikowana poprawnie. Algorytm trenowania SVM prowadzi do takiego wyniku.

Przekształcenia kernelowe W przypadku, gdy klasy są nieliniowo separowalne, tak jak na rysunku 2.8 stosuje się do maszyny wektorów wspierających przekształcenia kernelowe, dzięki którym klasyfikator operuje na danych w taki sposób jak gdyby zostały przetransformowane do innej przestrzeni - dzięki temu hiperpłaszczyzna jest w stanie odseparować od siebie próbki, które w oryginalnej przestrzeni nie były liniowo seperowalne. Inną interpetacją jest zaginanie płaszczyzny separującej w oryginalnej przestrzeni.

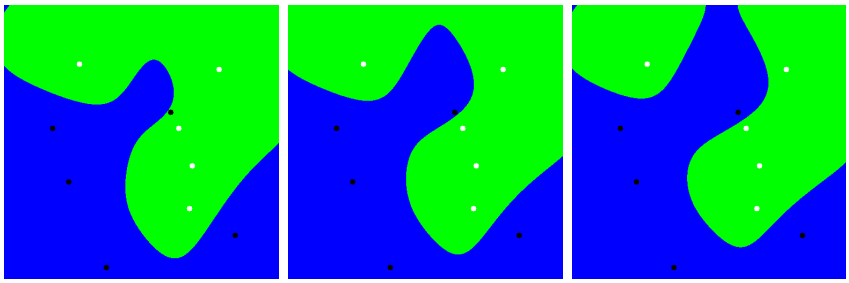
Najczęściej powszechnie stosowanym jest jądro normalne (Gaussowskie).

Parametry regulujące Z klasyfikatorem SVM i przekształceniami kernelowymi związane są dwa parametry regulujące:

- C - intuicyjnie: reguluje jak mocno hiperpłaszczyzna jest naginana aby dopasować się do próbek treningowych. Dzięki temu parametrowi jest możliwość ustalenia klasyfikatora w taki sposób, aby w mniejszym stopniu był dopasowany do danych treningowych, ale był bardziej ogólny
- γ - intuicyjnie: reguluje wpływ pojedynczej próbki treningowej.

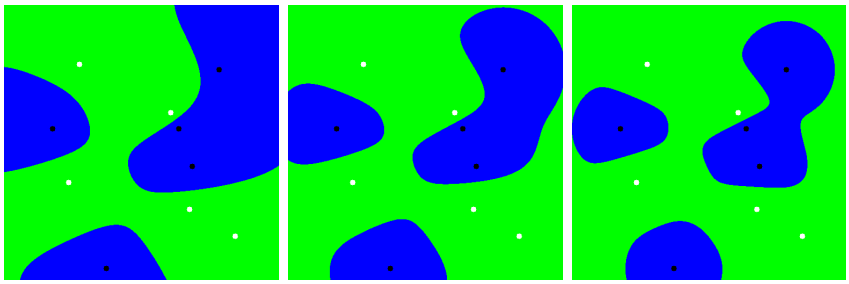
Rysunek 2.9 prezentuje wpływ parametru C na sposób wyznaczania klasyfikatora. Przy wyższych wartościach trenowany jest klasyfikator, który dobrze klasyfikuje wszystkie próbki treningowe - jest do nich bardzo mocno dopasowany. Przy niższych - część próbek treningowych zostałaby sklasyfikowana niepoprawnie, co oznacza, że klasyfikator jest gorzej dopasowany do danych treningowych, jest za to bardziej ogólny.

Rysunek 2.10 prezentuje wpływ parametru γ na sposób wyznaczania klasyfikatora. Kolorem czarnym oznaczone są próbki treningowe pierwszej klasy, białym - drugiej. Kolorem niebieskim - obszar na którym nowe obserwacje zostaną zaklasyfikowane do pierwszej klasy, zielonym - do drugiej.



(a) $C = \mathbf{2}$, $\gamma = 0.0001$ (b) $C = \mathbf{5}$, $\gamma = 0.0001$ (c) $C = \mathbf{10}$, $\gamma = 0.0001$

Rysunek 2.9: Wpływ parametru regulującego C na klasyfikator



(a) $C = 10$, $\gamma = \mathbf{0.0001}$ (b) $C = 10$, $\gamma = \mathbf{0.0002}$ (c) $C = 10$, $\gamma = \mathbf{0.0005}$

Rysunek 2.10: Wpływ parametru regulującego γ na klasyfikator

Klasyfikacja na wiele klas Jeśli zagadnienie polega na podziale obserwacji na **m** klas (gdzie, $m > 2$), stosuje się powszechnie jedną z dwóch metod:

1. $1+2+3+...+(m-1)$ klasyfikatorów, po jednym dla każdej pary klas. Każdy z nich decyduje o przynależności do jednej z dwóch klas. Aby sklasyfikować daną próbkę, odczytuje się wynik klasyfikacji każdego z nich, a następnie wybiera tą klasę, na którą wskazało najwięcej klasyfikatorów.

2. iteracyjnie m algorytmów SVM, gdzie każdy z nich klasyfikuje obserwacje na zasadzie: *należy do danej klasy* $\iff$ *nie należy.*

Sposoby oceny klasyfikatora Jednym z powszechnie stosowanych sposobów oceny klasyfikatora jest skuteczność klasyfikacji. Jego wartość odpowiada na pytanie: jaki procent danych treningowych został poprawnie zaklasyfikowany? Jest on definiowany poniższym równaniem:

$$\eta = \frac{N_c}{N_t} \times 100\% \tag{2.1}$$

gdzie

- η - skuteczność klasyfikacji
- N_c - liczba poprawnie sklasyfikowanych przykładów w zbiorze testowym
- N_t - liczba wszystkich przykładów w zbiorze testowym

2.3.3 Cechy sygnału

Współczynniki mel-cepstralne

Współczynniki mel-cepstralne (ang. *Mel Frequency Cepstrum Coefficients, MFCC*) - metoda obliczania krótkookresowych cech sygnału szeroko stosowana w wielu zagadnieniach związanych z uczeniem maszynowym. Przekształca sygnał (wektor z próbkami sygnału w kolejnych chwilach czasowych) w macierz, której kolumny odpowiadają kolejnym ramkom czasowym, zaś jej wiersze - kolejnym współczynnikom.

Sposób obliczania macierzy:

1. sygnał wejściowy jest dzielony na segmenty o określonej długości (mogą na siebie zachodzić)
2. każdy segment jest mnożony przez funkcję okna w celu zniwelowania efektu krawędzi
3. obliczana jest transformata Fouriera segmentu
4. obliczane są moce sygnału w kolejnych pasma częstotliwościowych należących do trójkątnych okien, których szerokości są równe w skali melowej
5. obliczona moc jest logarytmowana
6. na wyniku poprzedniej operacji, przeprowadzana jest dyskretna transformacja kosinusowa - każdy kolejny współczynnik tej transformaty to kolejny współczynnik MFCC danej ramki sygnału

Współczynniki SDC

Przesuwane współczynniki delta (ang. *Shifted delta coefficients, SDC*) - macierz cech obliczona na podstawie macierzy współczynników mel-cepstralnych.

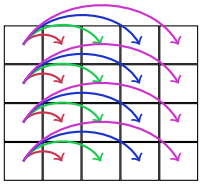
Sposób jej obliczania jest następujący: każda kolumna jest konkatenacją różnic między oddalonymi od siebie o kolejne odległości kolumnami macierzy MFCC.

Ilustruje to rysunek 2.11, na którym symbolicznie przedstawiona jest jedna z kolumn macierzy SDC, a kolorem zaznaczone są jej elementy, które odpowiadają różnicy między elementami macierzy MFCC oznaczonej strzałką w tym samym kolorze.

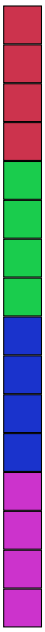
2.3.4 Modelowanie statystyczne

Model mieszanek gaussowskich

Model mieszanek gaussowskich (ang. *Gaussian Mixture Model, GMM*) - funkcja gęstości prawdopodobieństwa zdefiniowana jako ważona suma rozkładów normalnych (Gaussa). Ma zastosowanie przy modelowaniu zjawisk, których rozkład prawdopodobieństwa jest zbyt złożony aby był modelowany pojedynczym rozkładem normalnym.



(a) Macierz MFCC



(b) Kolumna macierzy SDC

Rysunek 2.11: Sposób obliczania macierzy jednej kolumny SDC na podstawie MFCC

Model ten jest jednoznacznie określony następującym zbiorem parametrów:

- wagi ω_i
- średnie μ_i
- macierze kowariancji Σ_i

gdzie i - indeks kolejnego pojedynczego rozkładu normalnego.

$$p(\boldsymbol{x}) = \sum_{i=1}^{M} \omega_i p_i(\boldsymbol{x}) \tag{2.2}$$

$$p_i(\boldsymbol{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu_i})'(\Sigma_i)^{-1}(\boldsymbol{x} - \boldsymbol{\mu_i}) \tag{2.3}$$

gdzie:

$p(\boldsymbol{x})$ - rozkład gęstości prawdopodobieństwa wystąpienia wektora $\boldsymbol{x}$

$p_i(\boldsymbol{x})$ - rozkład gęstości prawdopodobieństwa wystąpienia wektora $\boldsymbol{x}$ dla i-tego pojedynczego rozkładu

D - liczba wymiarów wektora obserwacji $\boldsymbol{x}$

Algorytm Expectation-Maximization

Algorytm pozwalający na wyznaczenie parametrów GMM na podstawie zbioru obserwacji, które opisują rozkład prawdopodobieństwa obserwacji.

Na wejściu przyjmuje dane treningowe (w przypadku niniejszej pracy zestaw macierzy MFCC/SDC obliczonych na podstawie zbioru nagrań) oraz oczekiwaną liczbę komponentów modelu mieszanek Gaussowskich, zwraca wszystkie parametry GMM wspomniane w poprzedniej sekcji.

Jego działanie polega na maksymalizacji logarytmicznej funkcji prawdopodobieństwa wszystkich danych treningowych w szukanym przez ten algorytm modelu mieszanek gaussowskich.

Funkcja prawdopodobieństwa wielu obserwacji w danym rozkładzie gęstości prawdopodobieństwa wyraża się jako iloczyn funkcji prawdopodobieństwa dla poszczególnych obserwacji (wzór 2.4). Powszechnie, w celu zoptymalizowania obliczeń oraz "kompresji" ich wyników, dla każdej obserwacji oblicza się logarytm funkcji prawdopodobieństwa, dzięki czemu iloczyn można zamienić na sumę (wzór 2.5).

$$p(\boldsymbol{D}) = \prod_{n=1}^{N} p(\boldsymbol{x_n}) \tag{2.4}$$

gdzie:

$\boldsymbol{D}$ - zbiór N obserwacji $\{\boldsymbol{x_1}, \boldsymbol{x_2}, ...\boldsymbol{x_n}\}$

n - numer kolejnej obserwacji

$p(\boldsymbol{x})$ - funkcja prawdopodobieństwa dla jednej obserwacji ze wzoru 2.2

$$\log p(\boldsymbol{D}) = \sum_{n=1}^{N} \log p(\boldsymbol{x_n}) \tag{2.5}$$

Uniwersalny model tła

Uniwersalny model tła (ang. *Universal Background Model, UBM*) - określenie modelu mieszanek gaussowskich wytrenowanego na podstawie dużej ilości danych, np. całego zbioru treningowego. W zagadnieniach związanych z rozpoznawaniem mówcy, nastroju itp. z fragmentu sygnału audio, UBM jest modelem mieszanek gaussowskich dla zbioru bardzo dużej liczby próbek dźwiękowych.

Adaptacja MAP

Adaptacja MAP (ang. *maximum a posteriori probability* - maksymalizacja prawdopodobieństwa a posteriori) - algorytm służący do adaptacji parametrów statystycznych obserwowanych danych do znanych parametrów uniwersalnego modelu tła.

Mając daną nową obserwację $\boldsymbol{X}$ (np. macierz cech MFCC i SDC sygnału audio) oraz uniwersalny model tła zawierający M komponentów, opisujący zbiór innych obserwacji (wielu innych nagrań), algorytm adaptacji MAP daje określony wzorem 2.6 zbiór wektorów zaadaptowanych wartości oczekiwanych $\{\boldsymbol{\mu_1^*}, \boldsymbol{\mu_2^*}, ..., \boldsymbol{\mu_M^*}\}$.

Każda z wartości oczekiwanych $\boldsymbol{\mu_i^*}$ modeluje statystycznie rozkład wartości i-tej kolumny macierzy cech (kolejne kolumny reprezentują kolejne ramki sygnału audio).

Wartość ta to ważona współczynnikiem α suma:

- wektora wartości oczekiwanych i-tego komponentu gaussa z modelu tła
- wektora uśrednionych kolejnych kolumn macierzy cech, przy czym liczona jest średnia ważona wartością gęstości prawdopodobieństwa danej kolumny dla i-tego komponentu rozkładu normalnego[1].

$$\boldsymbol{\mu_i^*} = \alpha \boldsymbol{E_i}(\boldsymbol{X}) + (1 - \alpha)\boldsymbol{\mu_i} \tag{2.6}$$

gdzie $\boldsymbol{E_i}(\boldsymbol{X})$ jest dane wzorem 2.7

$$\boldsymbol{n_i}(\boldsymbol{X}) = \sum_{t=1}^{T} Pr(i|\boldsymbol{x_t}) \tag{2.7}$$

$$\boldsymbol{E_i}(\boldsymbol{X}) = \frac{\sum_{t=1}^{T} Pr(i|\boldsymbol{x_t})\boldsymbol{x_t}}{\boldsymbol{n_i}} \tag{2.8}$$

gdzie $Pr(i|\boldsymbol{x_t})$ jest dane wzorem 2.9

$$Pr(i|\boldsymbol{x_t}) = \frac{\omega_i p_i(\boldsymbol{x_t})}{\sum_{j=1}^{M} \omega_j p_j(\boldsymbol{x_t})} \tag{2.9}$$

gdzie:

$$\alpha_i = \frac{n_i}{n_i + r} \tag{2.10}$$

r - współczynnik stosowności. Badania wykazały, że dla wartości 8-20, jego wartość nie ma istotnego wpływu na skuteczność adaptacji. W niniejszej pracy została więc przyjęta wartość 14.

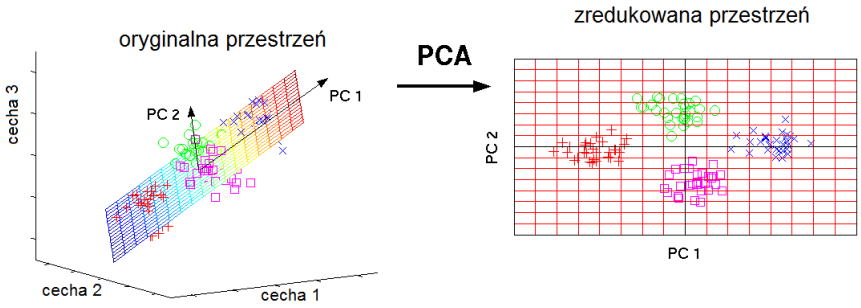
$\boldsymbol{X}$ - macierz cech obserwacji (np. nagrania)

$\boldsymbol{\mu_i^*}$ - zaadaptowana wartość oczekiwana odnosząca się do danej obserwacji (np. konkretnego nagrania) i i-tego numeru komponentu modelu tła

T - liczba segmentów sygnału

$\boldsymbol{x_t}$ - wektor reprezentujący cechy sygnału (np. współczynniki MFCC i SDC dla t-tego segmentu sygnału)

[1] W przypadku rozkładu jednowymiarowego (UBM z jednym komponentem), wartość $E_i(\boldsymbol{X})$ jest wektorem średnich arytmetycznych kolejnych kolumn macierzy cech.



Rysunek 2.12: Rysunek prezentujący zastosowanie algorytmu PCA[6]

Parametry należące do modelu tła:

M - liczba komponentów rozkładu normalnego

$p_i(x_t)$ - wartość funkcji gęstości prawdopodobieństwa dla wektora $\boldsymbol{x_t}$ w i-tym komponencie rozkładu normalnego

ω_i - waga i-tego komponentu rozkładu normalnego

$\boldsymbol{\mu_i}$ - wartość oczekiwana i-tego komponentu rozkładu normalnego

Superwektor

Pod tą nazwą w literaturze określany jest wektor będący konkatenacją wektorów $\boldsymbol{\mu_1^*}, \boldsymbol{\mu_2^*}, ..., \boldsymbol{\mu_M^*}$ z poprzedniej sekcji.

Każde nagranie o dowolnej długości, opisane przez macierze MFCC lub SDC, jest rzutowane na superwektor we wspólnej dla wszystkich próbek przestrzeni uniwersalnego modelu tła. Reprezentuje on statystyczny opis cech nagrania.

Jest on używany jako wektor cech na wejściu klasyfikatora.

2.3.5 Analiza głównych składowych

Analiza głównych składowych (ang. *Principal Component Analisys, PCA*) - metoda analizy zbioru danych należących do różnych klas w przestrzeni o n wymiarach, która pozwala znalezienie nowej n-wymiarowej przestrzeni, w której wariancje rozkładów próbek na kolejnych wymiarach są uszeregowane od największych do najmniejszych.

Pozwala to na zmniejszenie wymiarowości przestrzeni przy jak najmniejszej stracie informacji poprzez odrzucenie wymiarów o najniższych wariancjach.

Rysunek 2.12 ilustruje przykład, w którym wszystkie próbki znajdują się w niewielkiej odległości od dwuwymiarowej płaszczyzny znajdującej się w trójwymiarowej przestrzeni.

PCA w tym przypadku pozwala znaleźć tę płaszczyznę. Odrzucenie trzeciego wymiaru (projekcja punktów na znalezioną płaszczyznę) wiąże się z niewielką stratą informacji, pozwala za to na uproszczenie i przyspieszenie obliczeń związanych z punktami.

Rozdział 3

Opis systemu informatycznego

Aby odpowiedzieć na pytania z sekcji 2.2, zostało stworzone narzędzię - program komputerowy, który dokonuje przetworzenia danych oraz implementuje wspomniane w 2.3 algorytmy uczenia maszynowego.

Program posiada dwie funkcje:

1. Na podstawie danych treningowych oraz modelu tła tworzy superwektor, oraz trenuje klasyfikator SVM.
2. Przekształca zbiór muzyki w plikach MP3 w bazę danych z informacjami o sklasyfikowanym nastroju każdego utworu

Napisano także zbiór skryptów programu Matlab, których zadaniem było:

- Wczytanie utworów muzycznych ze zbioru, wstępna ich obróbka i przekształcenie w macierze cech
- Na podstawie macierzy cech obliczenie uniwersalnego modelu tła
- Na podstawie danych o wartościowości i pobudzeniu, przygotowanie pliku zawierającego klasy nastrojów przypisane do poszczególnych utworów, a także wyrównanie liczebności utworów należących do poszczególnych klas

Ich opis znajduje się w sekcji 4.1 opisującej procesy przeprowadzane przez te skrypty.

Została stworzona też **wtyczka do odtwarzacza multimedialnego Winamp**, która - mając do dyspozycji bazę danych - tworzy listę odtwarzania zawierającą utwory, których nastrój jest wybrany przez użytkownika.

3.1 Wtyczka do programu Winamp

Jest ona uaktywniana z poziomu odtwarzacza Winamp i jego celem jest ułożenie listy odtwarzania w tym programie w taki sposób, aby znalazły się na niej utwory o nastroju wybranym przez użytkownika. Informację o tym jaki nastrój posiada dany utwór, moduł pobiera z opisanej później bazy danych.

Rysunek 3.1: Okno Winampa wraz z wtyczką

3.1.1 Opis wyglądu i funkcjonalności

Wygląd programu

Rysunek 3.1 prezentuje wygląd wtyczki.

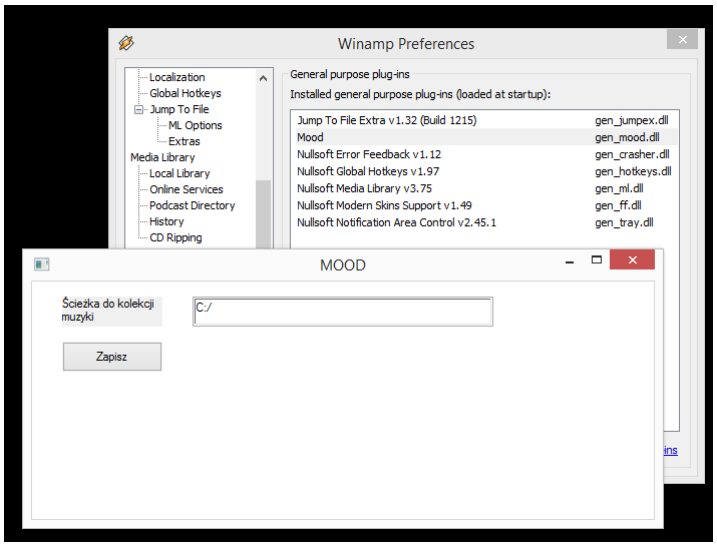
Funkcjonalność

Wybór nastroju Użytkownik wybiera nastrój muzyki, której chce słuchać poprzez kliknięcie jednego z 4 przycisków reprezentujących nastroje.

Program następnie wykonuje kolejno kroki:

1. Kasuje aktualną zawartość listy odtwarzania programu Winamp
2. Pobiera z bazy danych ścieżki plików z utworami, które mają wybrany nastrój
3. Dodaje te utwory do listy odtwarzania
4. W przypadku niepowodzenia, wyświetla komunikat o błędzie
   - *Brak plików MP3* - jeśli nie może znaleźć na dysku żadnego pliku wymienionego w bazie (w przypadku gdy brakuje części plików, są one omijane)
   - *Brak bazy* - jeśli nie znajduje pliku z bazą danych.
5. Uruchamia odtwarzanie

Budowanie bazy Po kliknięciu przycisku **stwórz bazę**, program uruchamia moduł trenowania i klasyfikacji z odpowiednimi ustawieniami, który skanuje pliki MP3, określa ich nastrój i zapisuje dane do bazy. W przypadku nie znalezienia tego modułu, wyświetla komunikat o jego braku.



Rysunek 3.2: Okno preferencji Winampa oraz okno konfiguracji wtyczki

Konfiguracja Program daje możliwość wyboru folderu, w którym są przechowywane pliki muzyczne. Aby ustawić ścieżkę tego folderu, należy otworzyć okno konfiguracji wtyczki (preferencje Winampa/konfiguracja plug-inu). Wygląd okna prezentuje rysunek 3.2. Użytkownik wpisuje tam ścieżkę, a następnie zapisuje ustawienie przez kliknięcie przycisku "zapisz".

3.1.2 Szczegóły implementacyjne i technologiczne

Moduł został napisany w języku C++ i skompilowany kompilatorem Microsoft Visual C++ 2013.

Komunikacja z Winampem

Aby aplikacja była wykrywana jako plug-in programu Winamp, musi spełniać kilka wymagań:

- Zostać skompilowana jako dynamicznie dołączana biblioteka (DLL)
- Posiadać odpowiednią nazwę. Wtyczki dotyczące biblioteki mediów (takie jak rozwijana w niniejszej pracy) powinny nazywać się ml_*.dll (gdzie * to dowolny ciąg znaków)
- Plik ze skompilowaną wtyczką powinien zostać umieszczony w folderze *Plugins* w katalogu głównym programu Winamp

Sterowanie Winampem z poziomu wtyczki Winamp udostępnia zestaw narzędzi *Winamp SDK*, w którego skład wchodzą między innymi pliki nagłówkowe, w których zdefiniowane są metody i zmienne umożliwiające komunikację wtyczki z programem.

Z poziomu wtyczki można uruchamiać różne metody Winampa. Służy do tego funkcja SendMessage() będąca częścią biblioteki WinAPI, które jest dostarczana wraz z systemem Windows i służy do komunikacji między programami i systemem operacyjnym.

Jako parametry należy podać tzw. *uchwyt* (rodzaj wskaźnika) do instancji programu, do której chcemy przesłać wiadomość. W tym przypadku był to uchwyt do uruchomionej aplikacji Winamp (podczas uruchamiania wtyczki, Winamp sam przekazuje jej wartość tego uchwytu). Należy też podać rodzaj wiadomości - identyfikator zdefiniowany w pliku *wa_ipc.h* z Winamp SDK.

Każda wiadomość może składać się z dwóch atrybutów (parametrów). W przypadku sterowania Winampem, jeden z nich to nazwa metody (włącz odtwarzanie, skasuj zawartość listy odtwarzania itp.), drugi - szczegółowe informacje dla tej metody, np. obiekt zdefiniowanej w pliku *wa_ipc.h* struktury zawierający informacje o utworach, które mają być dodane do listy odtwarzania.

Omawiany program wykorzystuje poniższe metody sterowania Winampem:

- uruchamianie odtwarzania (wciśnięcie przycisku play)
- czyszczenie listy odtwarzania (usunięcie wszystkich jej elementów)
- dodawanie utworu do listy odtwarzania

Implementacja graficznego interfejsu

Graficzny interfejs użytkownika został stworzony przy wykorzystaniu elementów biblioteki WinAPI.

Struktura bazy danych

Plik z bazą jest przechowywany w folderze AppData/Local/ml_mood w katalogu użytkownika w systemie Windows. Jest to plik tekstowy składający się z dwóch kolumn: jedna to numer nastroju (w zakresie 1-4), druga ścieżka pliku MP3.

Przykładowy fragment:

```
1 1 C:/Muzyka/Hedonism/My-Town.mp3
2 3 C:/Muzyka/Metallica/Enter-Sandman.mp3
```

Tworzenie bazy danych

Z poziomu wtyczki wywoływany jest (podczas jej uruchomienia) omówiony dalej moduł klasyfikacji. Powinien być zlokalizowany w folderze AppData/Local/ml_mood w katalogu użytkownika w systemie Windows. W tym samym folderze powinny znaleźć się pliki niezbędne do przeprowadzenia klasyfikacji: plik konfiguracyjny, plik z modelem klasyfikatora oraz plik z modelem tła.

Tabela 3.1: Argumenty wywołania aplikacji

-t	tryb trenowania (domyślnie: tryb klasyfikacji)
-c *nazwa pliku*	nazwa pliku z konfiguracją
-u *nazwa pliku*	nazwa pliku z UBM
-i *nazwa folderu*	nazwa folderu z plikami (XML lub MP3 w zależności od trybu)
-s *nazwa pliku*	nazwa pliku z modelem SVM
-p *nazwa pliku*	nazwa pliku z modelem PCA
-b *nazwa pliku*	nazwa pliku z bazą nastrojów i nazw plików (tryb klasyfikacji)
-m *nazwa pliku*	nazwa pliku z nastrojami (tryb trenowania)

3.2 Moduł trenowania i klasyfikacji

Moduł ten posiada dwa tryby pracy:

- tryb trenowania klasyfikatora
- tryb klasyfikacji utworów

Celem trybu trenowania jest wytrenowanie klasyfikatora SVM, jest on więc używany w fazie produkcji programu. Służy on też do doboru optymalnych parametrów pozostałych używanych algorytmów, ponieważ jednocześnie testuje skuteczność całego systemu.

Tryb budowania bazy jest używany przez użytkownika systemu za każdym razem kiedy buduje i uaktualnia bazę nastrojów.

3.2.1 Funkcjonalność

Komunikacja z użytkownikiem

Moduł jest programem pracującym w linii poleceń, który przyjmuje następujące parametry uruchomienia:

Przykładowe wywołanie:

```
1       E:\MoodRecognition\> moodClassifier.exe -t -c conf.xml -u
    ubm.xml -i Data -s svm.xml -p pca.xml -m moods.txt
```

Powoduje wytrenowanie przez program klasyfikatora na podstawie konfiguracji z pliku conf.xml, modelu tła z ubm.xml, nastrojów z pliku moods.txt i macierzy MFCC z folderu Data oraz zapisanie go w pliku svm.xml. Model PCA jest zapisywany w pliku pca.xml.

Pliki wejściowe i wyjściowe

W trybie trenowania

Program wczytuje pliki z danymi (macierzami cech), plik z metadanymi, plik z modelem tła oraz plik konfiguracyjny. Generuje dwa pliki wyjściowe: model SVM oraz plik, na podstawie którego można w programie GNU Plot rysować wykresy skuteczności klasyfikacji w zależności od dobranych parametrów.

Plik konfiguracyjny

W pliku konfiguracyjnym podawana jest liczba wymiarów wektora cech po redukcji z zastosowaniem algorytmu PCA. Parametr ten może przyjąć jedną wartość lub ich zakres. W przypadku zakresu, program dokonuje trenowania i testowania klasyfikatora SVM dla każdej wartości parametru.

Kolejne wartości parametru podawane są wewnątrz znaczników `<value>`, które znajdują się między znacznikami `<componentNumber>`.

```
<componentNumber>
    <value>1, 2, 5, 10, 20, 50</value>
</componentNumber>
```

W tym przypadku program stworzy superwektory oraz każdy z nich zredukuje w procesie PCA po kolei do 1, 2, 5, 10, 20 oraz 50 elementów.

Plik wejściowy z modelem tła Format XML. Plik wygenerowany przez funkcję obliczania modelu tła. Więcej w 4.2.1.

Pliki wejściowy z danymi treningowymi Pliki w formacie XML. Szczegółowy opis znajduje się w 4.1.

Plik wejściowy z nastrojami Numer klasy, która określa nastrój danego utworu ze zbioru treningowego.

Plik wyjściowy z modelem klasyfikatora Plik ten zawiera wszystkie parametry maszyny wektorów wspierających, które będą potrzebne przy klasyfikacji utworów przy użyciu tego samego klasyfikatora.

Plik wyjściowy z modelem PCA Plik ten zawiera parametry algorytmu analizy głównych składowych.

W trybie klasyfikacji utworów

Program wczytuje pliki typu MP3 z muzyką, plik z modelem tła, plik z danymi klasyfikatora oraz plik konfiguracyjny.

Plik wejściowy konfiguracyjny Ten sam plik, który został użyty do trenowania klasyfikatora.

Plik wejściowy z modelem tła Ten sam plik, który został użyty do trenowania klasyfikatora.

Plik wejściowy z modelem SVM Jest to plik, który został stworzony stworzony w trybie trenowania.

Plik wejściowy z modelem PCA Jest to plik, który został stworzony stworzony w trybie trenowania.

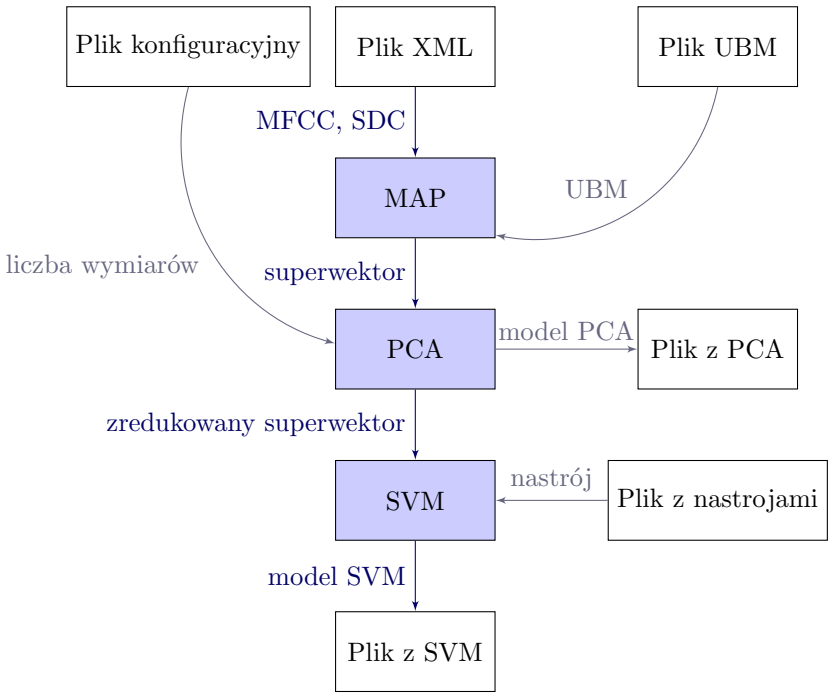
Plik wyjściowy z bazą nastrojów Plik tekstowy zawierający informacje o ścieżce pliku z utworem oraz jego nastrojem. Szczegółowy opis w 3.1.2.

3.2.2 Szczegóły technologiczne

Opis działania

Na początku, program sprawdza wprowadzone w linii poleceń parametry i podejmuje decyzje o tym, w którym pracuje trybie.

Trenowanie Rysunek 3.3 przedstawia proces trenowania.



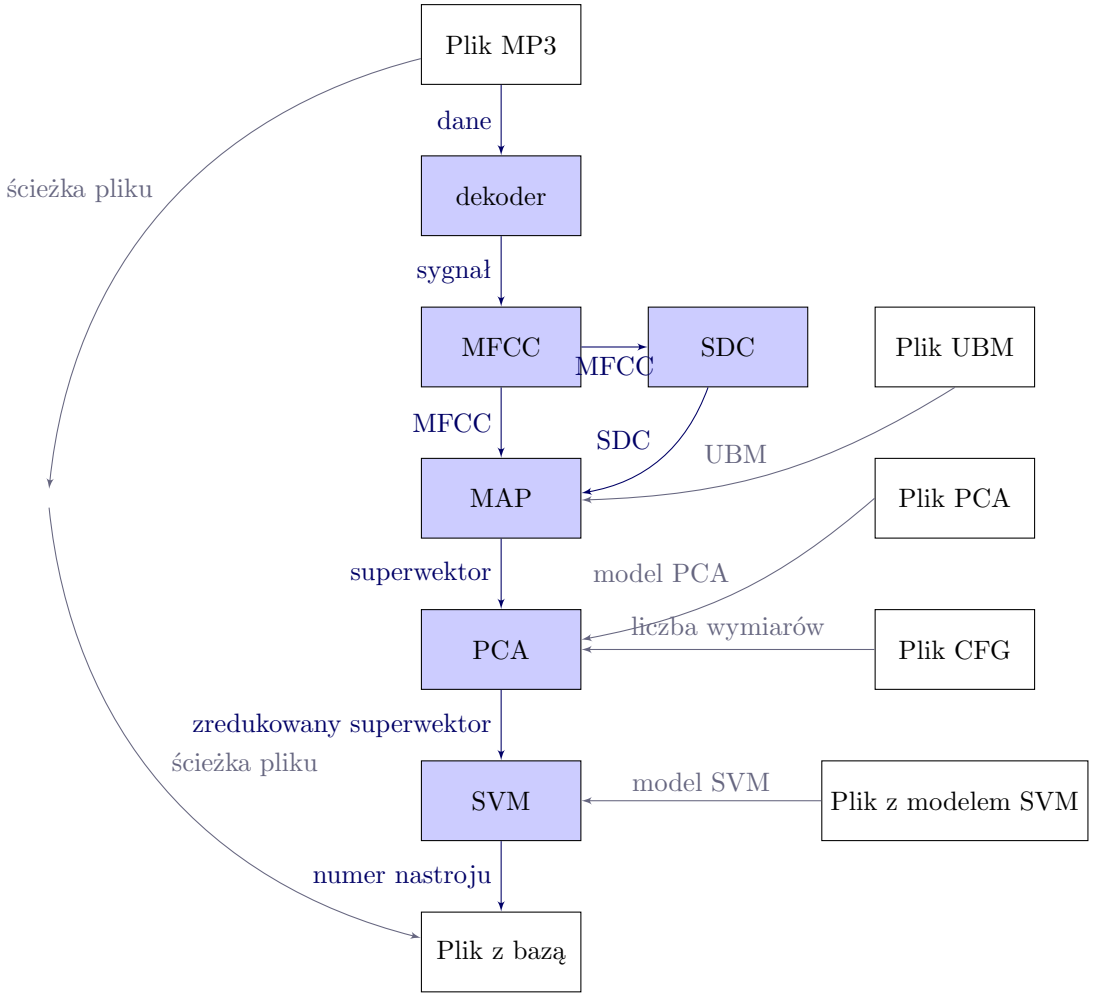
Rysunek 3.3: Schemat procesu trenowania klasyfikatora

1. wczytanie pliku z metadanymi (zawierającego informacje o nastroju każdego utworu ze zbioru treningowego)
2. wczytanie pliku z parametrami modelu tła oraz pliku konfiguracyjnego
3. przeskanowanie folderu z danymi treningowymi i zapisanie informacji (ścieżka i nastrój)

4. dla każdego pliku: odczytanie (z pliku *.xml) danych MFCC i obliczenie na ich podstawie elementów superwektora
5. analiza głównych składowych superwektora i redukcja jego długości
6. wytrenowanie klasyfikatora na podstawie wielu superwektorów i informacji o nastrojach utworów

Jeśli w pliku konfiguracyjnym, któryś z parametrów algorytmu MAP lub klasyfikatora zdefiniowany jest jako zakres, obliczanie superwektora i trenowanie odbywa się dla każdej z wartości z tego zakresu.

Budowanie bazy Rysunek 3.4 przedstawia proces budowania bazy.



Rysunek 3.4: Schemat procesu budowania bazy

1. wczytanie pliku z parametrami modelu tła oraz pliku konfiguracyjnego
2. rekurencyjne przeskanowanie folderu z muzyką (plików *.mp3) i zapisanie informacji o ścieżce

3. dla każdego utworu: zdekodowanie sygnału audio, ograniczenie go do pierwszych 30 sekund utworu, obliczenie na jego podstawie danych MFCC oraz SDC i obliczenie na ich podstawie elementów superwektora
4. analiza głównych składowych superwektora i redukcja jego długości
5. klasyfikacja superwektorów
6. zapisanie klas i ścieżek plików MP3 jako bazę danych

3.2.3 Szczegóły implementacyjne, użyte funkcje biblioteczne

PCA

Została użyta biblioteka OpenCV, w której skład wchodzi implementacja algorytmu PCA. Używane w pracy funkcje pochodzące z tej biblioteki to `PCA::project` - odpowiedzialna za wytrenowanie modelu PCA i `PCA::project`, której rolą jest przekształcenie wejściowego wektora do nowej przestrzeni.

Po transformacji do nowej przestrzeni, w której wariancja każdego kolejnego wymiaru jest mniejsza, następuje usunięcie informacji o dalszych, mniej znaczących, wymiarach. Liczba wymiarów do jakiej superwektor powinen być zredukowany jest wczytywana z pliku konfiguracyjnego.

SVM

Została użyta biblioteka OpenCV, w której skład wchodzi implementacja klasyfikatora SVM. Oferuje ona automatyczny dobór parametrów C i γ - szczegóły tego procesu opisane są w sekcji 4.2.3.

Użyte funkcje to `CvSVM::train_auto`, której zadaniem jest wytrenowanie klasyfikatora (wraz z doborem parametrów zbliżających klasyfikator do bycia optymalnym) oraz `CvSVM::predict`, która klasyfikuje wektor wejściowy na klasy.

Funkcja zamieniająca sygnał audio na macierz cech MFCC

Jest to funkcja napisana w języku programu Matlab, została więc na potrzeby niniejszej pracy przekształcona na kod w języku C++. Matlab oferuje moduł *Coder*, którego jedną z funkcji jest taka konwersja.

W oryginale funkcja używała wbudowanej w Matlaba funkcji `specgram()`, która ze względu na ograniczenia licencyjne, nie może zostać przetłumaczona na C++. Z tego powodu została zamieniona na funkcję `specgram()` z pakietu Octave. Zostało sprawdzone, że daje ona takie same wyniki jak funkcja oryginalna.

Rozdział 4

Przygotowanie danych, trenowanie klasyfikatora

4.1 Dane treningowe

Do trenowania i testowania systemu został użyty zbiór danych 1000 Songs Database [4]
Jest to zbiór fragmentów 744[1] utworów muzycznych w formacie MP3 wraz z ich nastrojami zawierający następujące gatunki:

- muzyka klasyczna
- folk
- country
- blues
- jazz
- rock
- pop
- muzyka elektroniczna

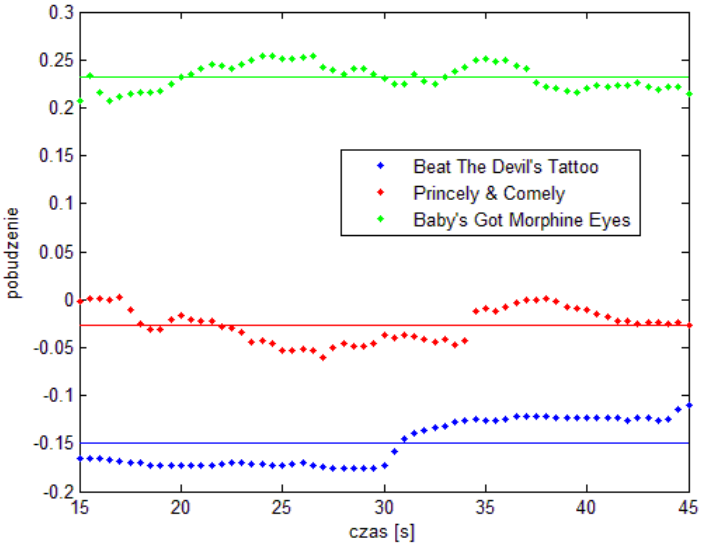
4.1.1 Przygotowanie danych treningowych

Reprezentacja nastroju w zbiorze danych

Z każdego utworu został przez autorów zbioru wybrany losowy fragment o długości 45 sekund, którego nastrój został określony przez ponad 300 pracowników serwisu Amazon Mechanical Turk. Mieli oni za zadanie osobno określić ich wartościowość i pobudzenie, zostali także poinstruowani aby kierować się nastrojem utworu jako takiego, a nie emocjami jakie utwór na nich wywołuje. Każdy z nich przeszedł wcześniej test kwalifikacyjny.

Nastrój był określany w sposób ciągły - tak więc każdy fragment jest reprezentowany przez zbiór jego wartości w dziedzinie czasu. Pierwsze 15 sekund zostało odrzucone, ostatecznie więc w bazie znajduje się funkcja nastroju dla 30 sekund.

[1] Różnica pomiędzy nazwą zbioru a liczbą utworów w nim zawartych wynika z faktu, że początkowo znajdowało się w nim 1000 próbek, twórcy odnaleźli jednak pośród nich duplikaty, które zostały następnie usunięte



Rysunek 4.1: Wykres pobudzenia i wartościowości w dziedzinie czasu

Rysunek 4.1 przedstawia wykresy pobudzenia (uśrednione dla wszystkich anotatorów) dla trzech przykładowych utworów: *Beat The Devil's Tattoo* z repertuaru zespołu Black Rebel Motorcycle Club, *Baby's Got Morphine Eyes* zespołu Zombie Prom Queen oraz *Princely and Comely* w wykonaniu The Agrarians.

Uśrednienie parametrów nastrojów W niniejszej pracy przyjęto (podobnie jak w większości literatury - por. sekcja 2.1), że nastrój jest niezmienny i opisuje cały fragment utworu, więc wektory te zostały uśrednione do jednej wartości (zaznaczone na wykresie 4.1 linią ciągłą).

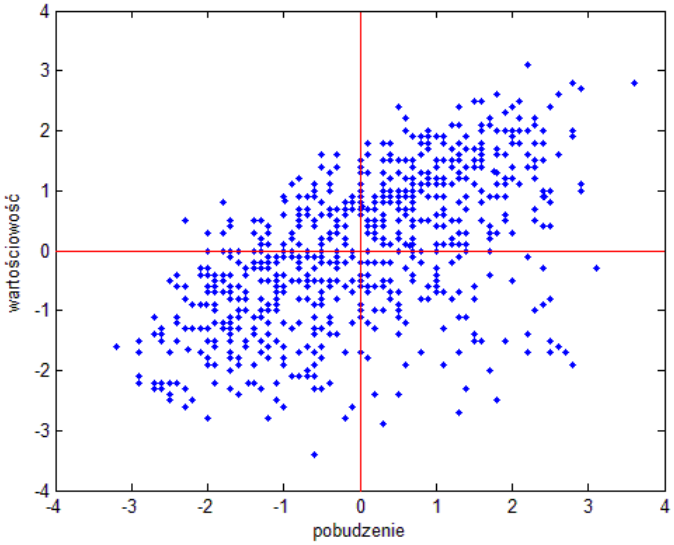
Rysunek 4.2 przedstawia rozkład średnich wartości wszystkich utworów z bazy na płaszczyźnie pobudzenie-wartościowość.

Mapowanie nastrojów na klasy Informacje o nastroju zostały odwzorowane na czterech klasach w zależności wartości parametrów pobudzenia i wartościowości. Tabela 4.1 prezentuje warunki jakie musiały spełniać parametry, aby próbka trafiła do danej klasy. Przyjęto, że decyzja zostaje podjęta po porównaniu parametru do zera.

W przypadku części próbek, wartość jednego z parametrów wynosiła 0. Strategia mapowania takich przypadków dla każdego parametru określona była w taki sposób aby możliwie maksymalnie wyrównać liczebność próbek w poszczególnych klasach.

Czerwone linie na wykresie 4.2 pokazują granice klas.

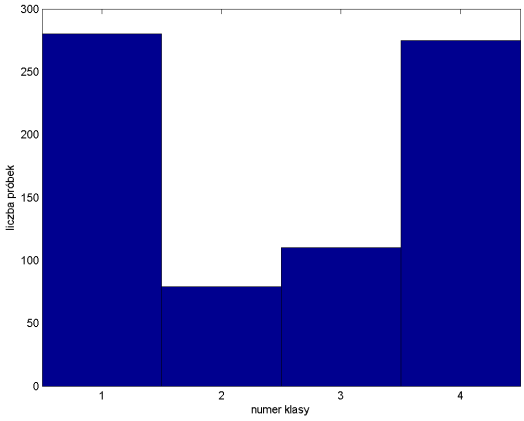
Wyrównanie liczebności klas Dane o nastrojach zostały przekształcone w taki sposób, aby liczba próbek treningowych (utworów) których wartościowość



Rysunek 4.2: Utwory ze zbioru treningowo-testowego na płaszczyźnie pobudzenie-wartościowość

Tabela 4.1: Strategia dzielenia próbek na klasy nastrojów

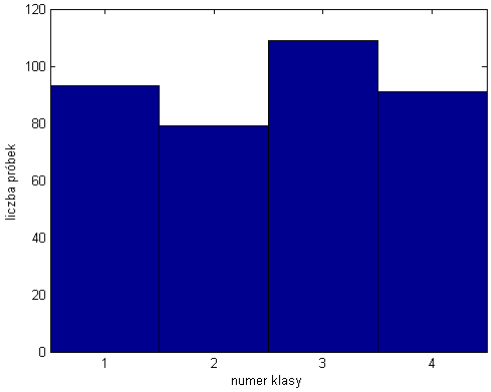
klasa		
1	pobudzenie $\leqslant 0$	wartościowość < 0
2	pobudzenie > 0	wartościowość $\geqslant 0$
3	pobudzenie $\leqslant 0$	wartościowość $\geqslant 0$
4	pobudzenie > 0	wartościowość < 0



Rysunek 4.3: Rozkład liczebności próbek w poszczególnych klasach

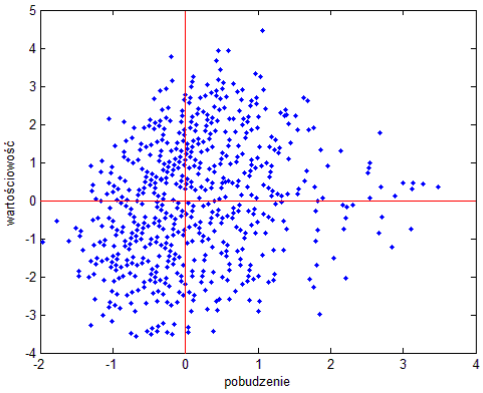
i pobudzenie są większe lub mniejsze od zera była zbliżona. W tym celu zostały wyznaczone mediany obu parametrów dla wszystkich próbek a następnie odjęte od każdej z nich. Liczebność próbek w poszczególnych klasach przedstawia histogram na rysunku 4.3.

Jak widać na histagramie rozkład ten nadal cechuje się bardzo dużą nierównomiernością - w klasach 1. i 4. znajduje się o wiele więcej próbek niż w pozostałych. Dlatego też w kolejnym kroku ze zbioru zostało usunięte $\frac{2}{3}$ utworów z tych dwóch klas. Rysunek 4.4 przedstawia rozkład po przekształceniach - liczebność próbek w każdej klasie jest zbliżona.



Rysunek 4.4: Rozkład liczebności próbek w poszczególnych klasach po wyrównaniu liczebności

Innym sposobem na wyrównanie rozkładu tego, byłby obrót przestrzeni przy użyciu algorytmu PCA (rysunek 4.5 obrazuje jak wtedy mógłby ten rozkład



Rysunek 4.5: Utwory ze zbioru treningowo-testowego po przekształceniu na płaszczyznę, na której liczebność klas byłaby wyrównana

wyglądać) - nie zostało to jednak zastosowane ponieważ doprowadziłoby do sytuacji, w której osie nie odwzorowałyby rzeczywistych wielkości pobudzenia i wartościowości, a pewne abstrakcyjne wielkości obliczone na ich podstawie.

Obliczenie macierzy współczynników MFCC oraz SDC

Dla każdego fragmentu została obliczona macierz współczynników MFCC przy pomocy funkcji MelFCC napisanej dla programu Matlab autorstwa Dana Ellisa[5] z parametrami przedstawionymi w tabeli 4.2.

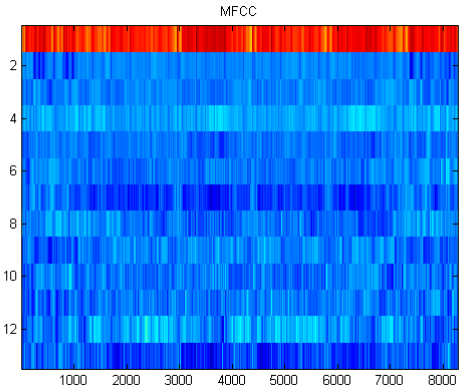
Tabela 4.2: Parametry funkcji MFCC

liczba współczynników	25
długość okna	0.0232 sekundy
długość zakładki	0.01 sekundy
częstotliwość dolna	0 Hz
częstotliwość górna	10 kHz
liczba filtrów melowych	40

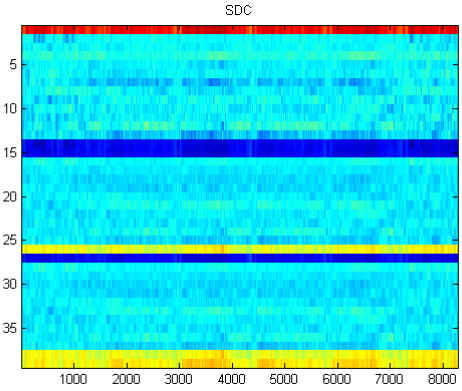
Na podstawie macierzy ze współczynnikami MFCC dla każdego utworu, zostały policzone współczynniki SDC przy użyciu funkcji mfcc2sdc napisanej dla programu Matlab autorstwa Md Sahidullaha [25] z parametrami przedstawionymi w tabeli 4.3.

Tabela 4.3: Parametry funkcji obliczającej SDC

odległość pomiędzy kolumnami które są od siebie odejmowane	1
odległość pomiędzy kolejnymi kolumnami, dla których współczynniki są obliczane	1
liczba obliczanych delt dla każdej kolumny	4



Rysunek 4.6: Współczynniki mel-cepstralne przykładowego utworu



Rysunek 4.7: Współczynniki SDC przykładowego utworu

Rysunek 4.6 przedstawia macierz cech MFCC dla utworu *Beat The Devil's Tattoo* z repertuaru zespołu Black Rebel Motorcycle Club, kolejny 4.7 przestawia macierz SDC.

Zapisanie macierzy MFCC i SDC do dalszych procesów Obliczone macierze zostały zapisane w dwóch formatach:

- w plikach MAT przechowujących dane programu Matlab - były one w dalszej kolejności wykorzystywane przez skrypt obliczający uniwersalny model tła
- w plikach XML - były używane przez program obliczający superwektory oraz trenujący i testujący klasyfikator

W obu typach plików znajdowały się dla każdego utworu te same dane. Osobno zostały zapisane macierze MFCC oraz konkatenacje macierzy MFCC z cechami SDC.

4.1.2 Usuwanie próbek zawierających wartości nienumeryczne

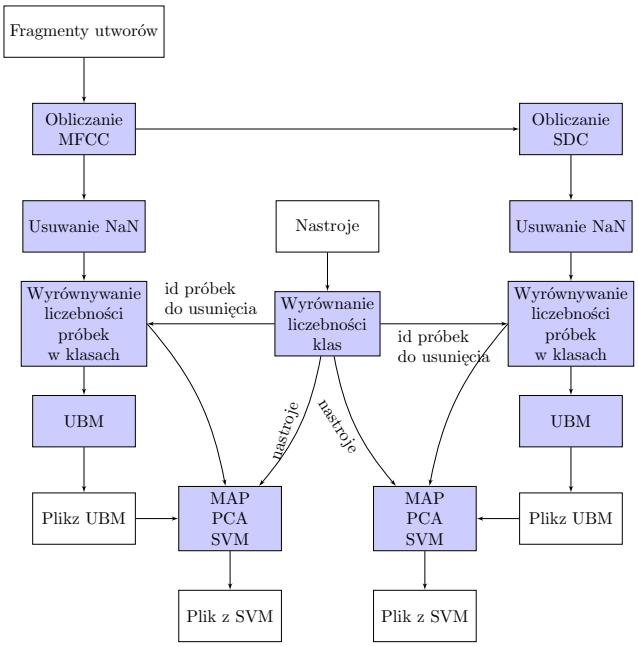
Jak zostało zauważone w dokumencie z zaleceniami dla uczestników konkursu MIREX [3], bardzo częstym problemem napotykanym przy przetwarzaniu rzeczywistych nagrań audio jest powstawanie wartości określanych jako *NaN, not a number* - nieliczbowych - spowodowanych ograniczoną precyzją obliczeń programów komputerowych. Autorzy dokumentu zwracają uwagę, aby programy uczestników były odporne na ten problem i zalecają odrzucenie takich próbek argumentując, że najlepszym rozwiązaniem jest zmniejszenie zbioru treningowego lub testowego o próbki powodujące takie problemy. W innym przypadku - jako, że liczby NaN powodują nieprzewidywane wyniki obliczeń, na przykład uniemożliwiają wytrenowania klasyfikatora - możliwa jest sytuacja, w której działanie całego systemu zostanie zaburzone.

W niniejszej pracy powyższe zalecenie zostało zastosowane i po każdym etapie obliczeń, próbki są sprawdzane pod kątem zawartości wartości NaN - wszystkie, które zawierają choć jedną są odrzucane. Dla różnych parametrów liczba takich próbek była w granicach 1% wszystkich, co nie wpływało w znaczący sposób na wielkość zbioru.

4.2 Opis procedury trenowania

Cała procedura wykonana była dwukrotnie: z uwzględnieniem wszystkich cech (MFCC oraz SDC) oraz bez cech SDC. Po zastosowaniu kroków opisanych w poprzedniej sekcji, zbiór testowo-treningowy zawierał 372 utwory muzyczne.

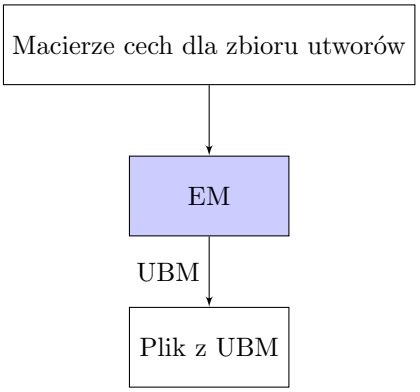
Rysunek 4.8 przedstawia proces trenowania.



Rysunek 4.8: Schemat procedury trenowania i testowania

4.2.1 Obliczanie uniwersalnego modelu tła

Rysunek 4.9 przedstawia proces trenowania UBM. Procedura trenowania została przeprowadzona na zbiorze treningowym przy użyciu funkcji programu Matlab `gmm_em` ze zbioru narzędzi MSR Identity Toolbox firmy Microsoft.



Rysunek 4.9: Schemat procedury trenowania UBM

Wybór liczby komponentów

Sposób obliczania teoretycznej liczby komponentów rozkładu normalnego przedstawia wzór 4.1.

$$k = \frac{n}{30l} \tag{4.1}$$

Tabela 4.4: Liczba komponentów rozkładu normalnego w modelu tła

	MFCC	MFCC+SDC
teoreryczne	1496	299
zastosowane w pracy	512	256

gdzie:

k - oczekiwaną liczbę komponentów

n - liczba danych (ramek wszystkich utworów)

l - liczba wymiarów

Tabela 4.4 przedstawia teoretyczne oraz przyjęte w pracy wielkości dla zbioru samych macierzy MFCC oraz dla zbioru połączonych cech MFCC i SDC. Liczbę komponentów w przypadku cech MFCC oraz SDC zmniejszono do 256, ponieważ użyta w pracy funkcja obliczająca model tła wymaga, aby liczba komponentów była potęgą liczby 2. W przypadku samej macierzy MFCC, mimo że obliczenia teoretyczne wskazują na użycie 1024 lub 2048 przyjęto liczbę 512 w celu przyspieszenia obliczeń - zarówno samego modelu tła, jak i - co bardzo ważne - superwektora[2].

Wybór liczby iteracji algorytmu EM

Zostało przyjęte kryterium stopu oparte na wystarczająco małym przyroście logarytmicznej funkcji prawdopodobieństwa (obliczanej dla zbioru treningowego) w kolejnej iteracji Expectation-Maximization. Graniczna wartość tego przyrostu została określona na podstawie obserwacji przebiegu wykresu pokazującego zależność między tą funkcją a numerem iteracji.

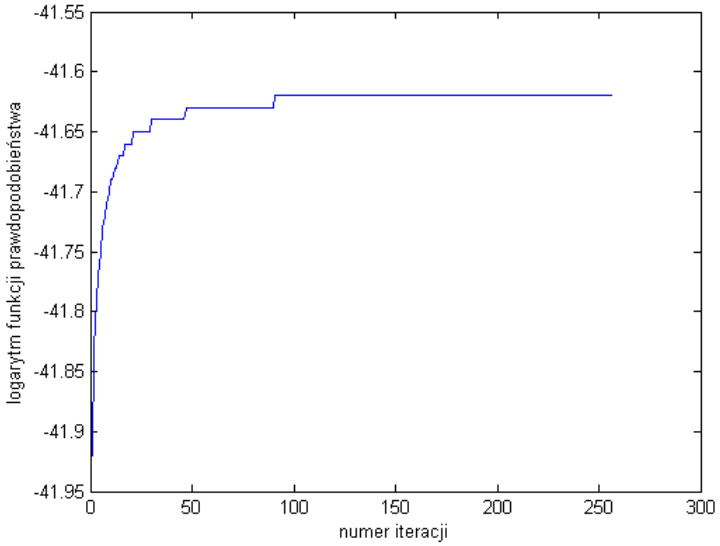
Rysunek przedstawia wykres zależności funkcji prawdopodobieństwa od liczby iteracji dla treningu danych zawierających tylko macierz MFCC przy 128 komponentach uniwersalnego modelu tła.

Wynika z niego, że wartość logarytmicznej funkcji prawdopodobieństwa nie zmienia się w sposób znaczący od iteracji numer 100. W podobny sposób określana była liczba iteracji przy dalszych obliczeniach, z innymi parametrami.

Wybór najlepszego modelu

Jako, że funkcja trenowania modelu mieszanek gaussowskich z używanej w pracy bibliotece działa w taki sposób, że inicjuje za każdym uruchomieniem te same parametry, każde jej użycie prowadzi do otrzymania dokładnie takiego samego modelu.

[2]Należy mieć na uwadze, że obliczenia superwektora wykonywane będą po stronie użytkownika



Rysunek 4.10: Zależność logarytmu funkcji prawdopodobieństwa od numeru iteracji

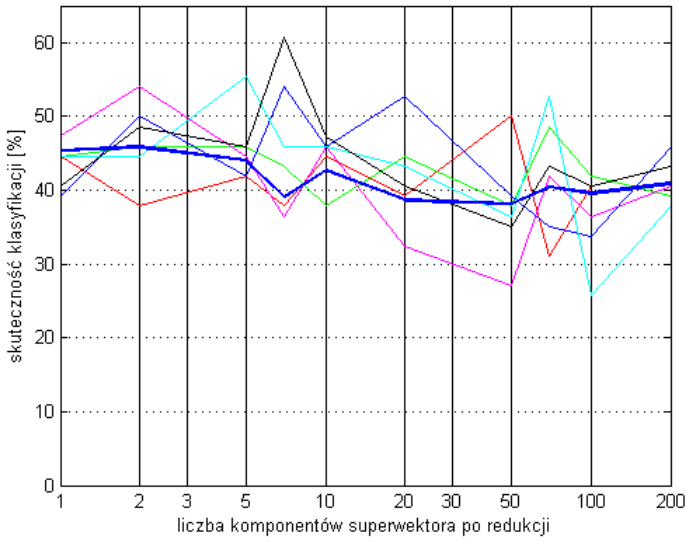
Lepszym podejściem byłoby inicjowanie losowych parametrach i wytrenowanie w ten sposób kilku modeli. Następnie należałoby określić który z nich najlepiej reprezentuje zbiór danych poprzez obliczenie logarytmicznej funkcji prawdopodobieństwa na podzbiorze testowym (nie używanym w treningu) i do dalszych procesów wybrać ten model, w przypadku którego ta funkcja osiągnie najwyższą wartość.

Zastosowanie powyższej metody jest rekomendowane przy dalszym rozwoju systemu.

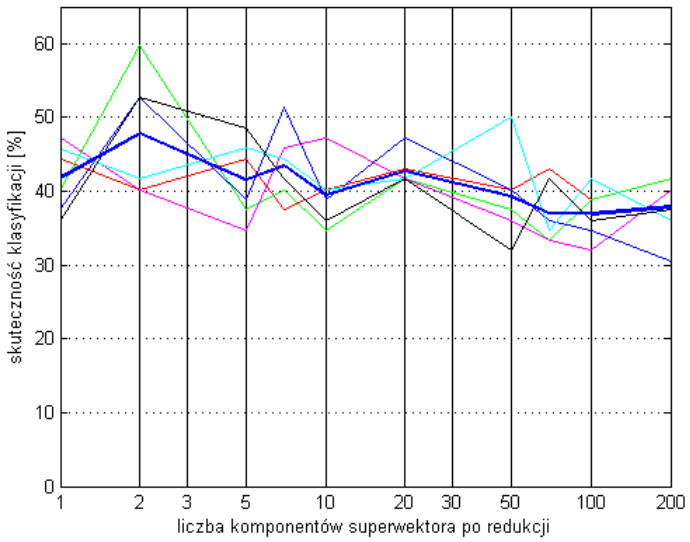
4.2.2 Dobór liczby wymiarów superwektora po redukcji w procesie PCA

Na podstawie obliczonych superwektorów został wytrenowany algorytm PCA. Następnie każdy superwektor został przekształcony do nowej przestrzeni przy użyciu tego algorytmu. Został też przeprowadzony test klasyfikacji z wyłączeniem algorytmu analizy głównych składowych - obliczony superwektor był bezpośrednio przekazywane do klasyfikatora SVM. Porównanie wyników tego testu z wynikami otrzymanymi przy użyciu PCA znajdują się w sekcji 4.2.4.

Moduł trenowania klasyfikatora został uruchomiony z konfiguracją określającą zakres liczby wymiarów superwektora po redukcji. Otrzymano wykres przedstawiający zależność pomiędzy tym parametrem a skutecznością klasyfikacji (rysunek 4.11 przedstawia tą zależność dla cech MFCC oraz konkatenacji MFCC i SDC. Są na nim zaznaczone wyniki otrzymane w sześciu uruchomieniach wraz z ich uśrednieniem - zaznaczonym grubszą linią).



(a) Tylko MFCC

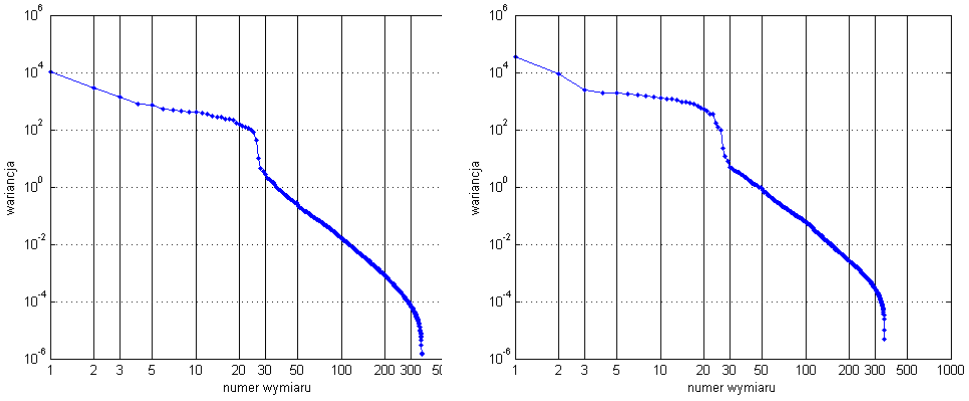


(b) MFCC oraz SDC

Rysunek 4.11: Skuteczność klasyfikacji w zależności od liczby wymiarów

Na wykresie można zaobserwować że skuteczność klasyfikacji nie wykazuje istotnej zależności od liczby wymiarów - podobne wyniki otrzymano zarówno dla przypadku jednego wymiaru i pięciuset. Szczegółowa analiza tych wyników wraz z komentarzem znajduje się w rozdziale 5.

Rysunek4.12 prezentuje wykres wariancji kolejnych elementów po analizie głównych składowych.



(a) Tylko MFCC (b) MFCC oraz SDC

Rysunek 4.12: Wariancja kolejnych wymiarów superwektora

W związku z powyższym, ustalono, że w systemie superwektor będzie redukowany do dwóch elementów. Podczas dalszych badań, po zmianie poszczególnych parametrów czy algorytmów należy jednak ponownie sprawdzić zależność skuteczności klasyfikacji od liczby wymiarów.

4.2.3 Trenowanie klasyfikatora

Została wybrana strategia: kilka klasyfikatorów, każdy decyduje o przynależności do jednej z dwóch klas.

Dobór parametrów

Na zbiorze treningowym zostało wytrenowane 6 klasyfikatorów podejmujących decyzję czy próbka należy do jednej z dwóch klas w sześciu parach. Przed treningiem każdego z nich, ze zbioru treningowego wydzielone zostały próbki należące do klas z pary.

Następnie został przeprowadzony dobór parametrów w procesie kroswalidacji. Dla każdego zestawu parametrów, zbiór treningowy był dzielony 5-krotnie na dwie części - podzbiór treningowy (4/5) i testowy (1/5). Klasyfikator był trenowany na każdym podzbiorze treningowym i testowany na odpowiadającym mu testowym z różnymi kombinacjami parametrów γ i C. Każdy test zwracał procentową skuteczność klasyfikacji. Z pośród 5 otrzymanych wartości najwyższa skuteczność determinowała ostateczny zestaw parametrów.

Tabela 4.6: Otrzymana skuteczność klasyfikacji

	bez PCA	z PCA
Tylko MFCC	40.54 %	46.85 %
MFCC + SDC	34.72 %	47.92 %
średnia	37.63 %	47.39 %

Otrzymane wyniki prezentuje tabela 4.5.

Tabela 4.5: Parametry klasyfikatorów

MFCC		
para klas	C	γ
1 vs 2	312.500	0.03375
1 vs 3	312.500	0.50625
1 vs 4	0.500	0.50625
2 vs 3	0.500	0.50625
2 vs 4	312.500	0.00225
3 vs 4	62.500	0.50625

MFCC + SDC		
para klas	C	γ
1 vs 2	0.100	0.50625
1 vs 3	2.500	0.50625
1 vs 4	312.500	0.50625
2 vs 3	12.500	0.03375
2 vs 4	12.500	0.50625
3 vs 4	12.500	0.50625

4.2.4 Walidacja skuteczności klasyfikacji utworów na nastroje

Każdy z dwóch zbiorów (osobny dla cech MFCC, osobny dla konkatenacji MFCC i SDC) obliczonych i zredukowanych superwektorów został sześciokrotnie podzielony w sposób losowy na zbiory testowy i treningowy w proporcjach 20% - 80%. Otrzymano w ten sposób zbiory o liczebności odpowiednio: 74 i 298 superwektorów.

Ostateczna skuteczność klasyfikacji została obliczona poprzez przetestowanie wyznaczonego w poprzednim kroku klasyfikatora na nieużywanym wcześniej do treningu zbiorze testowym - jest ona średnią skuteczności otrzymanych we wszystkich sześciu testach.

Otrzymano więc dwie wartości: dla wariantu z wszystkimi cechami oraz bez cech SDC. Wartości tych parametrów są przedstawione w tabeli 4.6.

Otrzymane wyniki są istotnie wyższe od losowego przypisywania klas (co dałoby skuteczność rzędu 25%).

Rozdział 5

Analiza podjętych działań i otrzymanych wyników

5.1 Ocena otrzymanych wyników

Wynik poprawności klasyfikacji jest niezadowalający w kontekście postawionego problemu, ponieważ - z punktu widzenia użytkownika - oznacza, że większość utworów będzie sklasyfikowana błędnie. Oznacza to, że w aktualnej fazie rozwoju, system będzie spełniał oczekiwania użytkowników lepiej niż powszechnie używany tryb tasowania, daleki będzie jednak od działania w pełni satysfakcjonującego.

Należy zwrócić uwagę na dwie własności wyników - bardzo wysokie odchylenie standardowe skuteczności klasyfikacji oraz brak zaobserwowanej zależności pomiędzy liczbą wymiarów superwektora po jego redukcji a skutecznością - zwłaszcza na fakt, że przy jednym wymiarze ma zbliżoną wartość do otrzymanej dla wyższych wymiarów. Można oczekiwać, że prawidłowo wytrenowany klasyfikator wykaże istotnie niższą skuteczność przy jednym wymiarze, a dla pozostałych - jego skuteczność nie będzie stała.

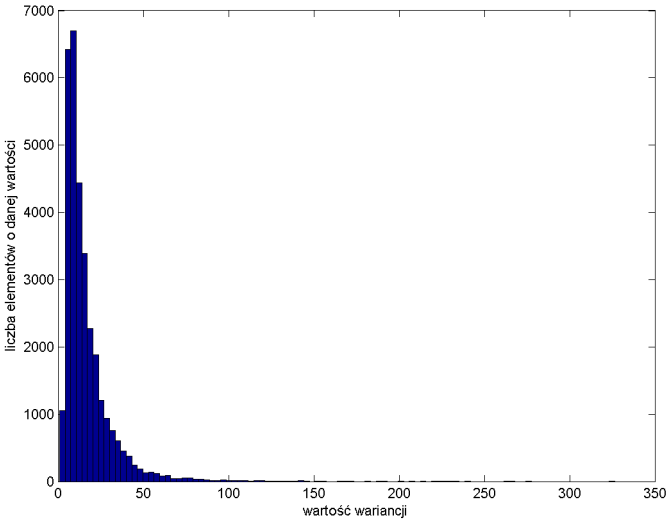
Wartość odchylenia standardowego skuteczności klasyfikacji wynosi 8.28 dla cech SDC i MFCC oraz 5.52 dla samych MFCC (przy zastosowaniu PCA i redukcji superwektora do dwóch wymiarów) W kontekście wartości średniej skuteczności klasyfikacji, daje to współczynniki zmienności na poziomie odpowiednio 0.17 oraz 0.12. Fakt, że wartości te są takie wysokie wynika z tego, że w każdym przebiegu procedury treningu i testu otrzymano skuteczność w istotnym stopniu różniącą się od pozostałych przebiegów. Wskazuje to na błędy powstałe na jednym z etapów całego procesu.

Jako, że liczebność klas w zbiorze treningowym i testowym była wyrównana, ponadto obydwa zbiory zawierały dość dużą (por. sekcja 2.1) liczbę próbek (372), przyczyną nie leży w niedopracowanym zbiorze.

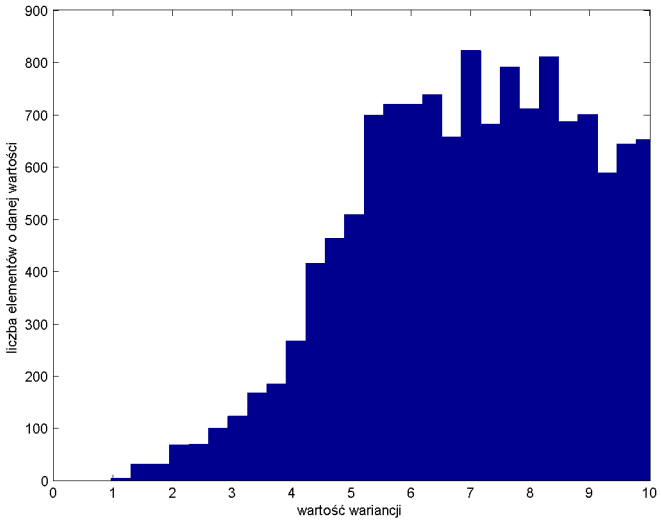
Źródłem problemu może być uniwersalny model tła, który może być przetrenowany. Przetrenowanie oznacza sytuację, w której jeden lub więcej kom-

ponentów modeluje cechy statystyczne jednego elementu. Kiedy taka sytuacja występuje, jej skutkiem jest bardzo niska wartość jednej z wag modelu tła lub niskie wartości znajdujące się na macierzy kowariancji.

Przeprowadzono analizę modelu tła pod kątem przetrenowania. Rysunek 5.1a przedstawia histogram wartości elementów wszystkich macierzy kowariancji w modelu. Na rysunku 5.1b histogram ten jest ograniczony do najniższych wartości. Jak widać, nie ma w żadnej z macierzy elementów, których wartość byłaby bliska lub równa 0. Rysunek 5.2 prezentuje podobny histogram rozkładu wartości wag w modelu. Jego analiza wskazuje, że żadna z wag nie jest bliska lub równa 0.



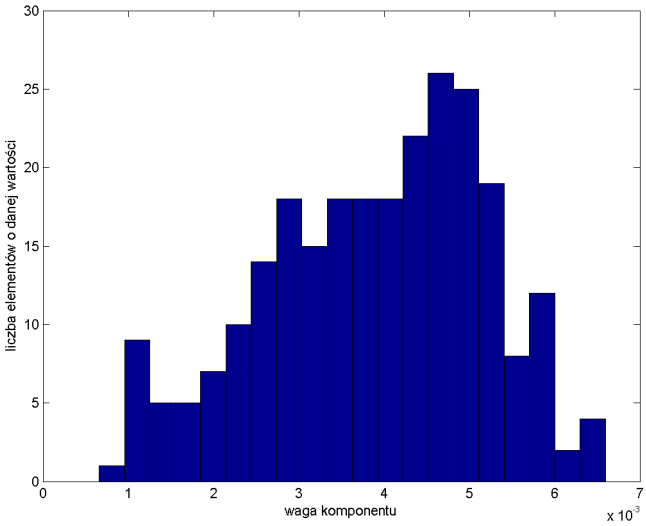
(a) Pełny zakres



(b) Zakres $< 0, 10 >$

Rysunek 5.1: Rozkład wartości elementów macierzy kowariancji w modelu tła

Powyższe fakty pozwalają przypuszczać, że źródłem problemu nie jest przetrenowanie uniwersalnego modelu tła. Należy jednak przeprowadzić testy z modelami, które zawierałyby mniejszą liczbę komponentów i dokonać obserwacji czy wpłynie to na poprawę działania klasyfikatora.



Rysunek 5.2: Rozkład wartości elementów macierzy kowariancji w modelu tła

Można przypuszczać, że przyczyną niedoskonałości wyników jest jeden z etapów procesu trenowania klasyfikatora SVM. Należy zbadać wytrenowany klasyfikator pod kątem przetrenowania. W przypadku klasyfikatora przetrenowaniem nazywa się sytuację, w której granica między klasami jest bardzo mocno dopasowana do jednej lub niewielkiej grupy próbek.

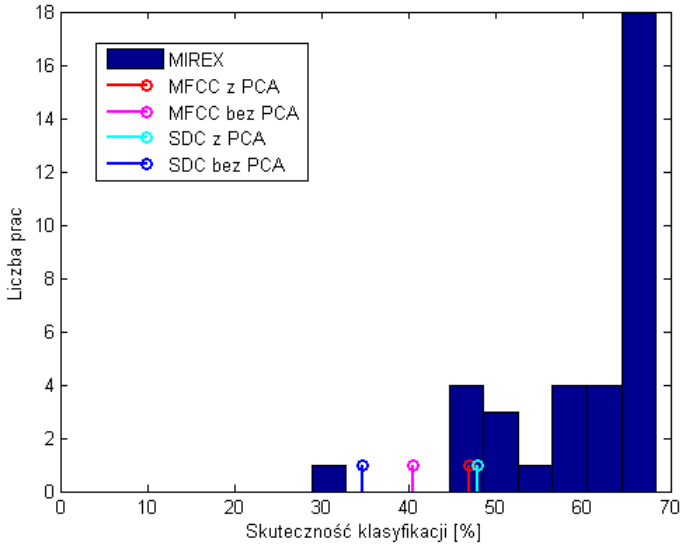
Należy przeanalizować wpływ parametrów regulujących C oraz γ na skuteczność klasyfikacji, jedną z przyczyn może być niewłaściwy dobór tych współczynników.

Porównanie wyników z innymi pracami z dziedziny rozpoznawania nastroju

Jak wynika z tabeli 2.1, skuteczność klasyfikacji algorytmów przysłanych na konkurs MIREX w latach 2014-2015 oscylowała wokół 60%. Część z nich wykazała jednak dużo niższą skuteczność - w zakresie 30% - 45%.

Mimo zauważonych niedoskonałości wyników klasyfikacji, postanowiono otrzymane wartości porównać z otrzymanymi przez autorów innych prac z dziedziny rozpoznawania nastroju. Porównanie to jest orientacyjne i rekomendowane jest jego powtórzenie po poprawieniu procesu klasyfikacji w celu otrzymania bardziej miarodajnego zestawienia wyników.

Rysunek 5.3 przedstawia pokazany wcześniej histogram 2.3 z zaznaczonymi wynikami otrzymanymi w ramach niniejszej pracy.



Rysunek 5.3: Rozkład skuteczność klasyfikacji pośród prac przysłanych na konkurs MIREX w latach 2013-2014 z zaznaczonym wynikiem otrzymanym w niniejszej pracy

5.2 Weryfikacja hipotezy o zwiększeniu skuteczności klasyfikacji poprzez uwzględnienie współczynników SDC

Jak wynika z tabeli 4.6, skuteczność klasyfikacji nie wykazuje istotnej różnicy pomiędzy przypadkiem użycia współczynników melcepstralnych oraz przypadkiem użycia zarówno współczynników MFCC i macierzy SDC w procesie parametryzacji.

W kontekście wspomnianych wad procesu klasyfikacji, obserwacja ta może być obarczona błędem, nie pozwala więc na potwierdzenie lub zaprzeczenie hipotezy.

5.3 Weryfikacja hipotezy o zwiększeniu skuteczności klasyfikacji gdy użyty jest algorytm analizy głównych składowych

Wyniki zawarte w tabeli 4.6 wskazują, że w obu przypadkach (zarówno cechy MFCC oraz MFCCi i SDC) wyższą skuteczność klasyfikacji otrzymano gdy analiza głównych składowych (PCA) oraz redukcja długości superwektora do elementów o najwyższej wariancji została uwzględniona.

Podobnie jak wyżej nie można jednoznacznie potwierdzić ani zaprzeczyć hipotezy na podstawie obserwacji.

5.4 Dalszy rozwój systemu

5.4.1 Poprawa skuteczności

Parametryzacja i klasyfikacja

Jednym ze sposobów poprawy skuteczności klasyfikacji może być zastosowanie, w procesie parametryzacji sygnału, cech wysokopoziomowych - związanych na przykład z tempem utworu lub jego tonacją.

Jak wspomniano w sekcji dotyczącej uniwersalnego modelu tła, rekomendowane jest zastosowanie biblioteki, która pozwala na wytrenowanie kilku różnych modeli, następnie ich walidacja na zbiorze testowym, po której następuje wybór modelu najlepiej pasującego do danych wejściowych.

Należy również rozważyć użycie różnego zestawu cech oraz różnych klasyfikatorów dla poszczególnych klas oraz podejmowanie ostatecznej decyzji na temat nastroju utworu po analizie wyników przez nie otrzymanych.

Zbiór treningowy

Jak zostało wspomniane wcześniej, rozkład klas w zbiorze treningowym był nierównomierny - liczebność próbek w niektórych klasach była znacznie większa niż w innych. W celu wyrównania, zbiór został znacznie zmniejszony. W procesie dalszego rozwoju systemu, należy przygotować zbiór, w którym liczebność próbek w poszczególnych klasach będzie wyrównana, dzięki czemu nie będzie potrzeby jego redukcji.

Należy także zastanowić się nad zastosowaniem innego sposobu określania nastroju nagrań. Jednym z możliwych rozwiązań, byłaby klasteryzacja cech utworów, tj. proces, w którym automatycznie wyznaczane są grupy próbek, których cechy wykazują podobieństwo, a następnie przy pomocy grupy słuchaczy odnalezienie związku pomiędzy otrzymanymi klastrami a odczuwanymi nastrojami. Dzięki temu klasy nastroju byłyby lepiej powiązane z parametrami utworów.

5.4.2 Poprawa wydajności

Zauważono, że najdłużej trwające procesy obliczeniowe to wyznaczanie modelu tła oraz obliczenia superwektorów. Jako, że model tła obliczany jest jednokrotnie, w fazie rozwoju programu, nie ma wpływu na działanie aplikacji uruchomionej przez użytkownika.

Z kolei proces obliczania superwektorów jest przeprowadzany każdorazowo podczas klasyfikacji nowych nagrań, czas jego trwania ma znaczący wpływ dla użytkowników. Należy więc dążyć do optymalizacji algorytmu adaptacji MAP. W tym celu proponowane jest użycie wielu rdzeni procesora. Jako, że większość komputerów domowych obecnie dysponuje dwoma lub czterema rdzeniami, pozwoli to na dwu- lub czterokrotne przyspieszenie obliczeń. Należy także rozważyć inne metody optymalizacji, w tym zmniejszenie precyzji obliczeń - w tym celu zalecane jest przebadanie wpływu precyzji na skuteczność klasyfikacji nastroju.

5.4.3 Poprawa funkcjonalności dla użytkownika

Zbudowany system zakłada, że do klasyfikacji nastroju utworu muzycznego brany jest pod uwagę fragment z początku kompozycji, a nastrój jest stały przez całe nagranie. Założenie to spełnia większość utworów z gatunku muzyki popularnej, są jednak wyjątki, których na przykład początek jest spokojny, końcówka zaś energiczna. Dalszy rozwój systemu powinien rozwiązać problem klasyfikacji takich przypadków.

Jedną z metod byłoby przypisywanie do nagrania dwóch cech - nastroju początkowego i końcowego. Program tworzący listę odtwarzania w takim przypadku mógłby dobierać nagrania w taki sposób, aby nastrój końca jednego był taki sam jak początku kolejnego. Należy jednak przeprowadzić dalsze badania opinii wśród słuchaczy i na ich podstawie wyciągnąć wnioski czy taka modyfikacja byłaby uzasadniona.

Rozdział 6

Podsumowanie

Tematyką pracy magisterskiej była automatyczna analiza nastroju nagrań muzycznych - zagadnienie wychodzące naprzeciw oczekiwaniom słuchaczy szukających sposobu na prosty i szybki dobór repertuaru do odsłuchu w zależności od chwilowej potrzeby. Omówione zostały trendy dominujące w ostatnich latach w dziedzinie elektroniki użytkowej, a zwłaszcza ułatwiania obsługi aplikacji multimedialnych. Przeprowadzone zostało jakościowe badanie opinii słuchaczy na temat ich potrzeb odnośnie wspomagania doboru repertuaru. W jego wynikach zaobserwowano zapotrzebowanie na oprogramowanie rozpoznające nastrój muzyki w sposób automatyczny.

Postawione zostały dwa cele: badawczy (omówienie, przetestowanie i porównanie metod wspomagających rozpoznawanie nastroju) oraz inżynieryjny - zaprojektowanie i zbudowanie kompletnego, przyjaznego użytkownikom, systemu wspomagającego dobór repertuaru.

Cel inżynierski osiągnięto - stworzono aplikację klasyfikującą przechowywane na dysku użytkownika utwory pod kątem nastroju oraz rozszerzenie do odtwarzacza multimedialnego udostępniające funkcję opracowującą listę odtwarzania na podstawie wybranego nastroju. Cały system został zbudowany w sposób pozwalający na jego rozszerzenie zarówno pod kątem użycia skuteczniejszych metod klasyfikacji i parametryzacji, jak i pod kątem ułatwiania dostępu do niego użytkownikom - między innymi poprzez budowanie rozszerzeń innych dostępnych na rynku odtwarzaczy muzycznych.

Cel badawczy został osiągnięty częściowo. Przygotowano wszystkie niezbędne do badań elementy - zbiory: treningowy i testowy wraz z przypisanymi poszczególnym ich elementom nastrojami oraz wytrenowane na ich podstawie modele: uniwersalny model tła, model analizy głównych składowych oraz maszynę wektorów wspierających. Wszystkie powyższe elementy zostały utworzone w dwóch egzemplarzach - dla dwóch metod parametryzacji sygnału dźwiękowego: przy użyciu współczynników MFCC oraz współczynników SDC wraz z MFCC.

Przeprowadzono porównanie wspomnianych dwóch metod parametryzacji, a także zbadano wpływ użycia algorytmu analizy głównych składowych na sku-

teczność klasyfikacji nastroju.

Na każdym etapie przetwarzania danych sformułowano zestaw uwag i obserwacji, które będą przydatne zarówno autorom kolejnych rozwiązań jak i przy dalszym rozwoju opisywanego w niniejszej pracy systemu.

Ostatecznie otrzymano wyniki wskazujące na niedoskonałości jednego z etapów przetwarzania. Wyniki te omówiono oraz przeprowadzono analizę dotyczącą możliwych przyczyn tych błędów, oraz sposobów na ich naprawę.

W aktualnej fazie rozwoju system jest gotowy do użytku przy bazującym na automatycznym rozpoznawaniu nastroju doborze repertuaru. Rekomendowane jest jednak prowadzenie zgodnie ze sformułowanymi zaleceniami dalszych badań nad poprawą jakości klasyfikacji nastrojów. Dziedzina nauki zajmująca się omawianym zagadnieniem jest stosunkowo nowa, a wyniki publikowanych prac naukowych wskazują, że należy rozwijać, testować i udoskonalać kolejne metody rozpoznawania emocji w nagraniach.

Systemy rozpoznające automatycznie nastrój należy rozwijać i ulepszać nie tylko pod kątem osiągania wysokiej skuteczności klasyfikacji emocji, ale także pod kątem optymalizacji oprogramowania - dążyć do zmniejszenia czasu potrzebnego na rozpoznanie nastroju, ponieważ jest to jeden z elementów wpływający na komfort użytkowania aplikacji.

W trakcie tworzenia systemu oraz przy przeglądzie literatury zaobserwowano, iż dużą trudnością jest znalezienie takiego sposobu opisywania nastroju utworu, że wszyscy słuchacze byliby zgodni co do oceny poszczególnych nagrań. Poza udoskonalaniem metod klasyfikacji, należy prowadzić badania pozwalające znaleźć lepsze sposoby określania nastrojów.

Spis rysunków

Spis tabel

Bibliografia

[1] 2014:audio classification (train/test) tasks - MIREX wiki. http://www.music-ir.org/mirex/wiki/2014:Audio_Classification_(Train/Test)_Tasks. Dostęp: 2014-07-03.

[2] ALLMUSIC - Frequently Asked Questions. http://www.allmusic.com/faq. Dostęp: 2014-05-06.

[3] Best coding practices for mirex. http://www.music-ir.org/mirex/wiki/2006:Best_Coding_Practices_for_MIREX. Dostęp: 2015-09-12.

[4] Emotion in music database (1000 songs). http://cvml.unige.ch/databases/emoMusic/. Dostęp: 2015-09-12.

[5] Plp and rasta (and mfcc, and inversion) in matlab using melfcc.m and invmelfcc.m. http://labrosa.ee.columbia.edu/matlab/rastamat/. Dostęp: 2015-08-30.

[6] Principal component analysis. http://www.nlpca.org/fig_pca_principal_component_analysis.png. Dostęp: 2015-06-30.

[7] James Bergstra, Michael Mandel, Razvan Pascanu, and Douglas Eck. LINEAR TAG PREDICTION FROM SPECTROGRAM COVARIANCE.

[8] Frédéric Bimbot, Jean-François Bonastre, Corinne Fredouille, Guillaume Gravier, Ivan Magrin-Chagnolleau, Sylvain Meignier, Teva Merlin, Javier Ortega-García, Dijana Petrovska-Delacrétaz, and Douglas A. Reynolds. A tutorial on text-independent speaker verification. *EURASIP journal on applied signal processing*, 2004:430–451, 2004.

[9] Juan José Burred and Alexander Lerch. A hierarchical approach to automatic musical genre classification. In *Proceedings of the 6th international conference on digital audio effects*, page 8–11, 2003.

[10] Juan Jose Burred, Mathieu Ramona, Frederic Cornu, and Geoffroy Peeters. Mirex-2010 single-label and multi-label classification tasks: ircamclassification09 submission. *MIREX 2010*, 2010.

[11] Karam Byun, Seung-Ryoel Baek, Jong-Soel Lee, Sei-Jin Jang, and Moo Young Kim. AUDIO TAG CLASSIFICATION: MIREX 2013 SUBMISSIONS.

[12] Chuan Cao and Ming Li. Thinkit's submissions for MIREX2009 audio music classification and similarity tasks. In *MIREX abstracts, International Conference on Music Information Retrieval*. Citeseer, 2009.

[13] Christophe Charbuillet, Damien Tardieu, Geoffroy Peeters, and others. Gmm supervector for content based music similarity. In *International Conference on Digital Audio Effects, Paris, France*, page 425–428, 2011.

[14] Franz De Leon and Kirk Martinez. USING TIMBRE MODELS FOR AUDIO CLASSIFICATION. 2013.

[15] A. Grecu and A. Rauber. MIREX 2010 POSTPROCESSING AUDIO DESCRIPTORS FOR IMPROVED AUDIO CLASSIFICATION.

[16] Xiao Hu and J. Stephen Downie. Exploring mood metadata: Relationships with genre, artist and usage metadata. In *ISMIR*, page 67–72. Citeseer, 2007.

[17] Xiao Hu, J. Stephen Downie, Cyril Laurier, Mert Bay, and Andreas F. Ehmann. The 2007 MIREX audio mood classification task: Lessons learned. In *ISMIR*, page 462–467. Citeseer, 2008.

[18] Youngmoo E. Kim, Erik M. Schmidt, and Lloyd Emelle. MoodSwings: A collaborative game for music mood label collection. In *ISMIR*, volume 8, page 231–236, 2008.

[19] Cyril Laurier. *Automatic Classification of Musical Mood by Content Based Analysis*. Universitat Pompeu Fabra, 2011.

[20] Tao Li and Mitsunori Ogihara. Detecting emotion in music. In *ISMIR*, volume 3, page 239–240, 2003.

[21] M. Mandel and D. Ellis. LABROSA's audio music similarity and classification submissions. *Music Information Retrieval Information Exchange (MIREX)*, 2007.

[22] Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based organization and visualization of music archives. In *Proceedings of the tenth ACM international conference on Multimedia*, page 570–579. ACM, 2002.

[23] Yannis Panagakis and Constantine Kotropoulos. Automatic music mood classification via low-rank representation. In *Proc*, page 689–693, 2011.

[24] Petr Pollak and Martin Behunek. Faculty of electrical engineering, czech technical university in prague, technická 2, 166 27 prague, czech republic. In *Signal Processing and Multimedia Applications (SIGMAP), 2011 Proceedings of the International Conference on*, page 1–6. IEEE, 2011.

[25] Md Sahidullah. Shifted delta coefficients (sdc) computation from mel frequency cepstral coefficients (mfcc). `http://www.mathworks.com/matlabcentral/fileexchange/31478/`. Dostęp: 2015-09-03.

[26] Klaus Seyerlehner, Markus Schedl, Peter Knees, and Reinhard Sonnleitner. DRAFT: A REFINED BLOCK-LEVEL FEATURE SET FOR CLASSIFICATION, SIMILARITY AND TAG PREDICTION.

[27] Ming-Ju Wu. MIREX 2013 submissions for train/test tasks.

[28] Yi-Hsuan Yang, Yu-Ching Lin, Ya-Fan Su, and Homer H. Chen. A regression approach to music emotion recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(2):448–457, 2008.