



POLITECHNIKA POZNAŃSKA

Łukasz Wieczorek

Rozpoznawanie i cyfryzacja ręcznie tworzonej notacji muzycznej

Praca magisterska

Promotor: dr inż. Adam Wojciechowski

Poznań 2014

Dziękuję promotorowi
za cenne wskazówki

Spis treści

Spis treści	1
1 Wstęp	3
1.1 Wprowadzenie	3
1.2 Cel i zakres pracy	7
2 Historia muzyki, edytory notacji muzycznej oraz oprogramowanie do optycznego rozpoznawania muzyki	9
2.1 Historia muzyki	9
2.2 Popularne edytory notacji muzycznej	16
2.3 Popularne oprogramowanie do optycznego rozpoznawania muzyki . .	23
3 Projekt programu	31
3.1 Badanie znajomości programów do edycji notacji muzycznej	31
3.2 Funkcjonalność programu i wybrane oprogramowanie	36
3.3 Architektura	37
3.4 Diagram przypadków użycia	38
4 Implementacja programu	39
4.1 Proces rozpoznawania symboli	39
4.2 MuseScore	43
4.3 Testy	43

5 Podsumowanie i wnioski	47
Bibliografia	49
Spis rysunków	51
Spis tablic	53
A Przewodnik użytkownika	55
A.1 Instalacja	55
A.2 Obsługa	56

Rozdział 1

Wstęp

1.1 Wprowadzenie

Pomysł na pracę magisterską o powyższym tytule zrodził się po wykonaniu zadania programistycznego związanego z programem MuseScore, dotyczącego odtwarzania fragmentu pliku reprezentującego partyturę w zadany sposób (powtarzanie, podnoszenie/obniżanie skali w kolejnych powtórzeniach). Jako że MuseScore jest programem o otwartym kodzie źródłowym, łatwo dodawać swoje modyfikacje. Wynikiem zadania była modyfikacja kodu MuseScore umożliwiająca wspomniane zadania. Niedługo po wykonaniu zadania autor zaczął się interesować nowościami proponowanymi przez branżę dostarczającą na rynek oprogramowanie dla muzyków. W wyniku rozmów z programistami MuseScore autor wybrał jako cel następnego zadania zbadanie możliwości usprawnienia procesu wprowadzania nut.

Z problemem wprowadzania notacji muzycznej spotyka się na swojej drodze większość muzyków, którzy chcą podzielić się, utrwały lub odtworzyć wybraną partyturę. Popularne programy do edycji notacji muzycznej reprezentują trzy główne podejścia do jej wprowadzania: podejście wizualne, podejście klawiaturowe i podejście tekstowe. Podejścia te oprócz swoich zalet mają również wady. W podejściu wizualnym większość czasu traci się na wskazywanie wysokości oraz pozycji wstawianych elementów. W podejściu klawiaturowym należy najpierw opanować metodę wprowadzania, która jest zwykle różna pomiędzy programami. Podejście tekstowe wymaga natomiast dobrej umiejętności pisania na klawiaturze oraz opanowania języka opisu

notacji muzycznej. W poniższej pracy zaprezentowane zostanie inne podejście, które jest z historią zapisu notacji muzycznej najbardziej związaną, mianowicie wprowadzanie poprzez pisanie nut z użyciem tabletu.

Współczesna notacja nutowa powstała na skutek modyfikacji notacji menzuralnej w I połowie XVII wieku. „Współczesny kształt nut, których elementami są główka zaczernione lub białe i pionowe laski, często z tak zwanymi chorągiewkami lub też poziomymi kreskami łączącymi grupę nut jest modyfikacją i rozwinięciem znaków graficznych używanych w notacji menzuralnej białej w XVI wieku,” [9]. Obecnie używane nuty wyglądają następująco:

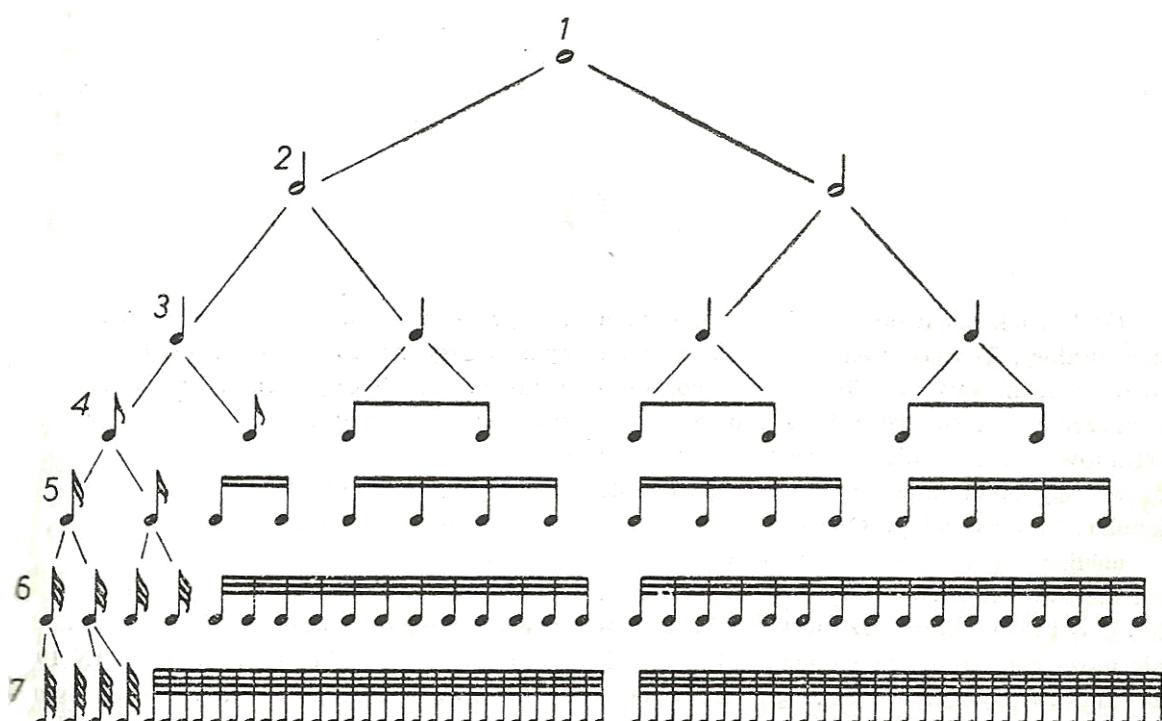
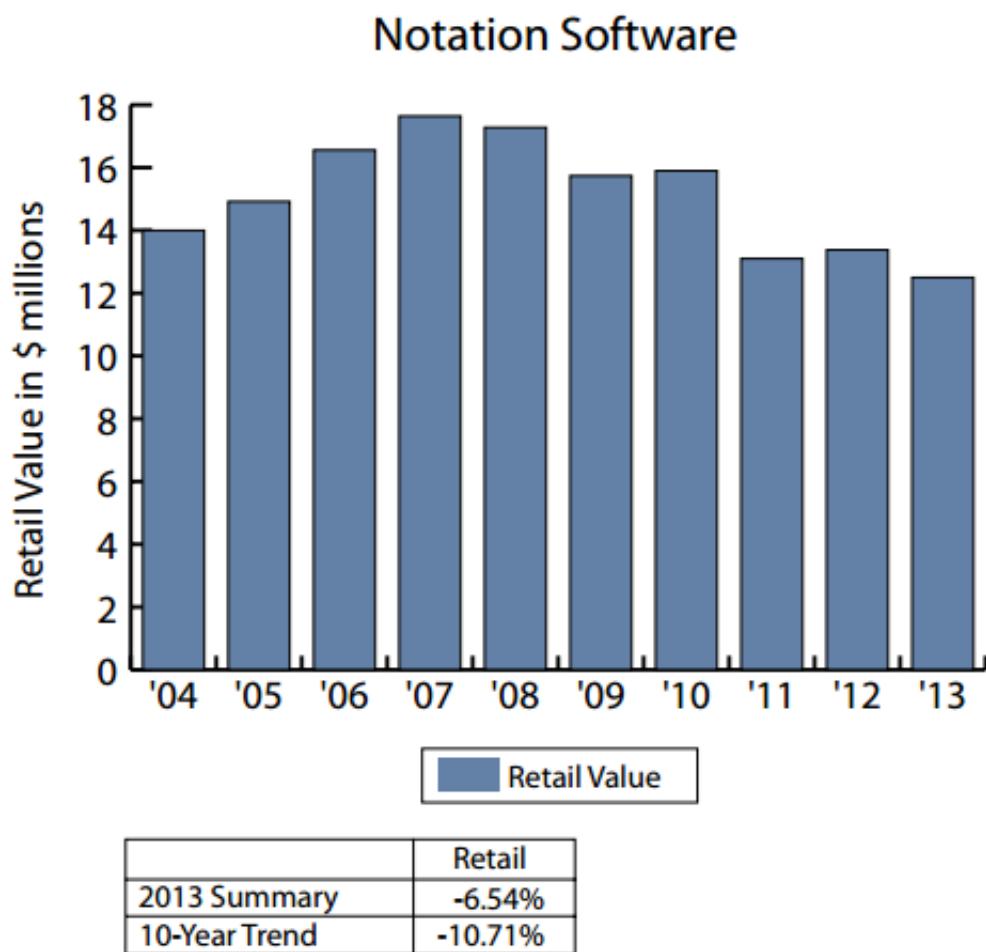


Tabela nut: 1) cała nuta, 2) półnuta, 3) ćwiartka, 4) ósemka, 5) szesnastka, 6) trzydziestodwójkowa, 7) sześćdziesięcioczwórka.

Rysunek 1.1: Tabela nut używanych obecnie [9]

Najpopularniejsze obecnie programy do edycji nut to: Finale, Sibelius. Z darmowych rozwiązań wyróżniają się LilyPond oraz MuseScore. W przypadku MuseScore dostępne są informacje na temat ilości pobrań programu z serwisu SourceForge (<http://sourceforge.net/>). W momencie pisania tekstu było to ponad 5,6 miliona pobrań [13] od roku 2004. Obecnie MuseScore korzysta z innego serwisu do udostępniania plików, z dostępnych danych [14] wynika, że w 2014 roku było ponad

0,6 miliona pobrań. Niestety pozostali producenci oprogramowania nie udostępniają dokładnych informacji na temat ilości sprzedanych licencji. Dostępny jest natomiast roczny raport stowarzyszenia NAMM (National Association of Music Merchants) dotyczący sprzedaży w Stanach Zjednoczonych Ameryki. Zgodnie z raportem NAMM z roku 2014 sprzedaż oprogramowania do edycji notacji muzycznej spadła o 6,5% w stosunku do roku poprzedniego, tj. do poziomu 12,5 miliona dolarów [22].



Rysunek 1.2: Raport stowarzyszenia NAMM z roku 2014 dot. oprogramowania do edycji notacji muzycznej [22]

LilyPond jest raczej przeznaczony dla doświadczonych użytkowników komputerów ze względu na tekstowe wprowadzanie danych. Pozostałe wymienione programy cechują się interfejsem udostępniającym wprowadzanie danych za pomocą myszki, klawiatury i innych rozwiązań, dlatego z pewnością są przeznaczone dla szerszej audiencji. Potwierdza to ocena w rankingu TopTenReviews [16], gdzie Sibelius oraz

Finale otrzymały noty 10 i 9 odnośnie do łatwości użycia w 10 stopniowej skali. W badaniu dotyczącym MuseScore z 2013 roku większość użytkowników zapytanych o łatwość użycia programu odpowiadała, że jest „zadowolona”, bądź „bardzo zadowolona” [15].

Pomimo wielkiej popularności zapisu nutowego istnieją także inne systemy notacji pomagające utrwalić dzieła muzyczne. Najpopularniejszym z nich z pewnością jest system tabulatury dla instrumentów strunowych (gitara, bass, etc.). Składa się on z linii reprezentujących struny oraz liczb, wskazujących próg na instrumencie. Poniżej reprezentacja zawierająca tabulaturę oraz odpowiadający zapis nutowy.

The image shows two measures of sheet music for guitar. The key signature is one sharp (F#). Measure 25 starts with a treble clef and a '25' above it. It consists of six eighth-note chords: (B,D,G,B), (D,F,A,D), (G,B,D,G), (B,D,G,B), (D,F,A,D), and (G,B,D,G). Each chord is preceded by a vertical bar line and followed by a black horizontal bar below the strings. Measure 26 starts with a '26' above it. It contains six eighth-note chords: (D,F,A,D), (G,B,D,G), (B,D,G,B), (D,F,A,D), (G,B,D,G), and (B,D,G,B). The notation is identical to measure 25, with vertical bar lines and black horizontal bars indicating string muting.

Rysunek 1.3: Tabulatura gitarowa i odpowiadający zapis nutowy

Co ciekawe istnieje także system notacji dla niewidomych oparty na kodzie Braille'a. Poniżej pokazane są podstawowe tony dla różnych długości dźwięku.

Rysunek 1.4: Notacja muzyczna zapisana w kodzie Braille'a [2]

Więcej systemów bazujących na podstawowej notacji muzycznej można odnaleźć na stronie projektu The Music Notation Project [25].

Dziwić może fakt, że niektórzy kompozytorzy zatrudniają ludzi do przepisywania napisanych przez siebie nut do postaci cyfrowej. Wynika z tego, że podejście pisania ręcznego jest preferowane przez osoby, które piszą najwięcej partytur. W poniższej pracy sprawdzono, czy hybrydowe podejście, to jest wprowadzanie nut z tabletu za pomocą pisania ręcznego ma sens, czy też jest to nikomu niepotrzebny dodatek do istniejących rozwiązań.

1.2 Cel i zakres pracy

Celem pracy jest opracowanie procedur i konstrukcja narzędzi służących do rozpoznawania i cyfryzacji ręcznie zapisanej notacji muzycznej. W zakres pracy wlicza się następujące zadania:

- Dobór i integracja narzędzi do rozpoznawania i cyfryzacji ręcznie sporzązonej notacji muzycznej,
- Przygotowanie zestawów danych treningowych i testowych,
- Przeprowadzenie i dokumentacja testów,
- Ankietowe badanie znajomości narzędzi edycji notacji muzycznej wśród społeczności użytkowników MuseScore i LilyPond.

Po krótkim wprowadzeniu, w rozdziale 2. pracy poruszone zostały kwestie historyczne oraz opisane zostały edytory notacji muzycznej, oraz rozwiązania do optycznego rozpoznawania muzyki. Dalej, w rozdziale 3., przedstawiony został projekt programu DrawNotes. Rozdział 4. to szczegółowy opis implementacji programu DrawNotes. Pracę kończy krótkie podsumowanie, wnioski, podręcznik użytkownika oraz wykaz odwołań literaturowych.

Historia muzyki, edytory notacji muzycznej oraz oprogramowanie do optycznego rozpoznawania muzyki

2.1 Historia muzyki

Czym właściwie jest muzyka? W „Słowniczku muzycznym” czytamy: „sztuka piękna przebiegająca w czasie, której materiałem są dźwięki i inne odgłosy (np. szmery), wydobywane przez głos ludzki i różnego rodzaju instrumenty muzyczne i posiadające określoną głośność, czas trwania oraz sposób wykonania, następujące po sobie pojedynczo lub we współbrzmieniach w myśl pewnej prawidłowości melodyki harmonii i kontrapunktu, ujęte w prawidłowość metrorytmiczną i określoną całość formalną, zgodnie z wymaganiami estetyki” [21]. W innym źródle czytamy: „Muzyka jako jedna ze sztuk pięknych działa na naszą psychikę za pomocą dźwięków. Materiałem więc, z którego tworzy swe dzieła, utwory muzyczne, są dźwięki uporządkowane według pewnych zasad, charakterystycznych dla poszczególnych okresów rozwoju muzyki. Na dzieło muzyczne, będące organizmem, składa się [...]: rytm, melodia, harmonia, dynamika, agogika, barwa dźwięku, wreszcie budowa formalna (forma) i wykonanie. [...] Do podstawowych elementów zaliczamy rytm, melodię i harmonię” [18].

J. W. Reiss ujmuje historię muzyki w następujące fazy [10]:

- a) starożytny Wschód i starożytna Hellada (religia, poezja),
- b) średniowiecze (chorał gregoriański),
- c) rozkwit polifonii w XV i XVI wieku (chanson, madrygał),
- d) okres monodii dramatycznej (opera, oratorium, kantata),
- e) okres rokoko (sonata, symfonia, koncert),
- f) koryfeusze muzyki klasycznej: Haydn, Mozart, Beethoven,
- g) okres romantyzmu: Weber, Schubert, Chopin,
- h) neoromantyzm,
- i) impresjonizm,
- j) muzyka nowoczesna.

W centrum zainteresowania niniejszej pracy znajduje się notacja muzyczna, tj. „system znaków pisanych odzwierciedlających konstrukcję dzieła muzycznego i umożliwiających jego zapisanie i odtworzenie. [...] Najstarsze systemy notacyjne były związane z muzyką monodyczną i posługiwały się znakami z pisma niemuzycznego (znaki zaczerpnięte z alfabetów, z pisma ideograficznego i cyfry). Odzwierciedlały one, w sposób mało precyzyjny, wysokość dźwięków, a w niektórych przypadkach również rytm, ujmując jedynie ogólny schemat utworu, będący podstawą improwizacji. Najwcześniejszymi systemami notacyjnymi tego typu były notacja muzyczna chińska [...] (1300 p. n. e.); notacja muzyczna babilońska (800 p. n. e.), związana z pismem klinowym; nadto indyjskie systemy notacyjne. [...] Również grecka notacja, gdzie rozwinęto oddzielny system dla muzyki wokalnej i instrumentalnej, wywodziła się z liter alfabetu. [...] Zbliżona do tej notacji była średniowieczna notacja muzyczna arabska (X–XIII wiek), używająca liter alfabetu na oznaczenie wysokości oraz później wprowadzonych cyfr ujmujących relacje rytmiczne. O podobnym systemie wspominają również źródła perskie z XV wieku. [...] Europejska muzyka liturgiczna posługiwała się notacją muzyczną neumatyczną, której geneza nie jest dokładnie znana. Pierwotna cheironomiczna forma tej notacji nie określała ani absolutnej wysokości dźwięków, ani też rytmu, oddając jedynie ogólny zarys melodii i jej skomplikowaną strukturę akcentową i artykulacyjną. Notacja cheiro-

nomiczna zastąpiona została (ok. 1000) notacją diastematyczną, określającą precyjnie wysokość dźwięków przez umieszczenie neum na liniaturze, nadal jednak nieujmującą stosunków rytmicznych. [...] Z chwilą wykształcenia techniki organalnej notacja neumatyczna nieokreślająca ściśle rytmu została zarzucona na korzyść notacji modalnej, która rozwinęła się w szkole Notre Dame ok. 1175 ze znaków notacji romańskiej ujętych w system schematów rytmicznych. Głosy podpisywano wówczas nad sobą w sposób partyturowy. [...] W okresie frankońskim zerwano z systemem partyturowym na rzecz układu głosowego. Systematyzacji zdobyczy tych czasów dokonał po 1300 Philippe de Vitry, ujmując wzajemne relacje wartości w ramach modus, tempus i prolatio, stwarzając tym samym podstawy notacji menzuralnej, która przetrwała w swych ogólnych formach do 1600 i była bezpośrednim punktem wyjścia współczesnej notacji muzycznej. Menzuralne systemy notacyjne służyły muzycze wykonywanej zespołowo. Muzykę solową notowano w tabulaturach, posługując się w tym celu odrębną notacją, zależną od rodzaju instrumentu. Zasadniczo istniały 2 ogólne typy tabulatur: klawiszowe oraz lutniowe. [...] W ostatnich latach [...] obserwuje się zjawisko intensywnego tworzenia nowych form zapisu, zdolnych utrwać i przekazać wykonawcy zjawiska muzyczne niemieszczące się w możliwościach tradycyjnej notacji, dostosowanej do wymogów systemu dur-moll” [9].

Ważną zmianą dla systemów notacji muzycznej było wprowadzenie linii, tj. zastosowanie systemu liniowego — „w średniowiecznej notacji muzycznej neumy notowano początkowo bez linii; dwie stałe linie: żółtą dla dźwięku c i czerwoną dla f, wprowadził Guido z Arezzo (XI wiek); [...] układ pięcioliniowy przyjął się obecnie jako jedyny” [21]. Rozwinięciem idei linii jest wprowadzenie kluczy, których kształt zmieniał się często co kształty nut. Klucz to „znak graficzny umieszczany na początku każdej pięciolinii, wyznaczający dokładnie położenie i nazwę jednej nuty odpowiadającej dźwiękowi określonej wysokości” [21]. Poniżej na rysunku 2.1 oraz 2.2 widzimy zestawienia zmian pisowni kluczy.

WIEK	WIOLINOWY	ALT.-TENOR	BASOWY
	G	C	F
XIII	G	C	F
XV	G	C	F
XVII	G	C	F
XIX	G	C	F

Rysunek 2.1: Klucze muzyczne na przestrzeni wieków [18]

a) klucz g



b) klucz f



c) klucz c

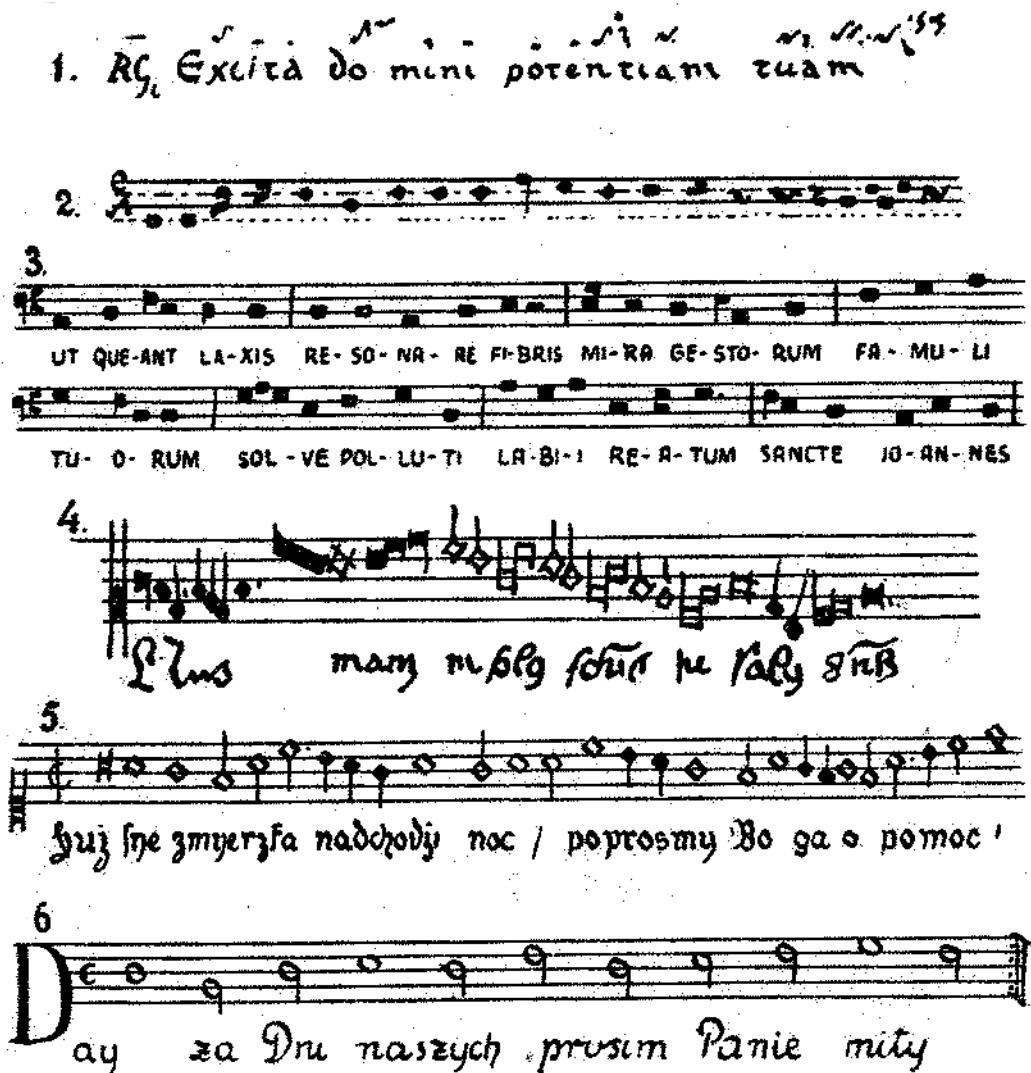


Rysunek 2.2: Klucze muzyczne na przestrzeni wieków cd. [9]

Wydawać by się mogło, że tabulatury to nowoczesny wynalazek, nic bardziej mylnego. Na rysunkach 2.3 oraz 2.4 widzimy tabulatury z XVI wieku. Tabulatury wykorzystują „specjalne systemy znaków, liter i cyfr” [21].

Rysunek 2.3: Tabulatura lutniowa z XVI wieku [18]

Rysunek 2.4: Tabulatura organowa z XVI wieku [18]



Rysunek 2.5: Pismo nutowe na przestrzeni dziejów. (1. Neumy z IX wieku. 2. Neumy z X–XI wieku. 3. Kwadratowe neumy rzymskie. 4. Pismo z XV wieku. 5. Pieśń z XVI wieku. 6. Pieśń XVII wieku.) [18]

Na rysunkach 2.6, 2.7, 2.8 oraz 2.9 przedstawiono neumy, notację modalną, notację menzuralną, oraz późniejszą notację włoską. Neuma w języku greckim znaczy skinienie, znak. Neumy definiuje się jako „prymitywne znaki średniowiecznej notacji muzycznej, której podstawowymi elementami była kreska (virga) i kropka (punctum); inne neumy są kombinacją tych znaków [...]; neumy, nieokreślające dokładnie wysokości i czasu trwania dźwięków, w różnych krajach i różnych okresach przybierały różną postać, stąd neumy bizantyjskie, włoskie, francuskie, gotyckie itp.” [21]. Kolejną w historii notacją jest notacja modalna — „kwadratowa notacja, system notacyjny wykształcony na gruncie muzyki wielogłosowej ostatniej czwierci

XII i I połowy XIII wieku. Celem notacji modalnej było ustalenie wartości rytmicznych [...] Źródłem notacji modalnej były starogreckie stopy metryczne zwane modi (stąd nazwa notacji)” [9]. Notacja menzuralna stosowana w XV wieku to „pismo nutowe późnego średniowiecza; rozwinęło się równolegle z rozwojem polifonii, w której ścisłe określenie wartości czasowej dźwięków było niezbędne” [21]. Notacja włoska natomiast to „odmiana notacji menzuralnej XIV wieku, spotykana głównie we włoskich rękopisach tego okresu, odznaczająca się daleko idącą prostotą systemu notacyjnego. Podstawową jednostką rytmiczną była w niej brevis, która była wartością niezmienną, w odróżnieniu od notacji francuskiej” [9].

	punctum		
	virga		brevis
	podatus albo pes		longa
	clivis albo clinis		modus trocheiczny
	porrectus		modus jambiczny
	torculus resupinus		modus daktyliczny
	scandicus flexus		modus anapestyczny
	climacus resupinus		modus spondeiczny
			modus trybrachyczny

Rysunek 2.6: Notacja neumatyczna (neumy) [10]

Rysunek 2.7: Notacja modalna [10]

	maxima (duplex long)	=	brevis
	longa		semibrevis
	brevis		semibrevis caudata
	semibrevis		semibrevis minima
	minima		semibrevis punktowana
	semiminima		
	fusa		
	semifusa		

Rysunek 2.8: Notacja menzuralna [9]

Rysunek 2.9: Notacja włoska [9]

Obecnie najpopularniejszym systemem notacji muzycznej jest zapis nutowy. Nuta to „znak graficzny w piśmie nutowym, określający wysokość dźwięku (zależnie od jego położenia na pięciolinii) oraz jego wartość rytmiczną (zależnie od kształtu)” [21]. Kształty nut zostały pokazane na rysunku 1.1 na stronie 4.

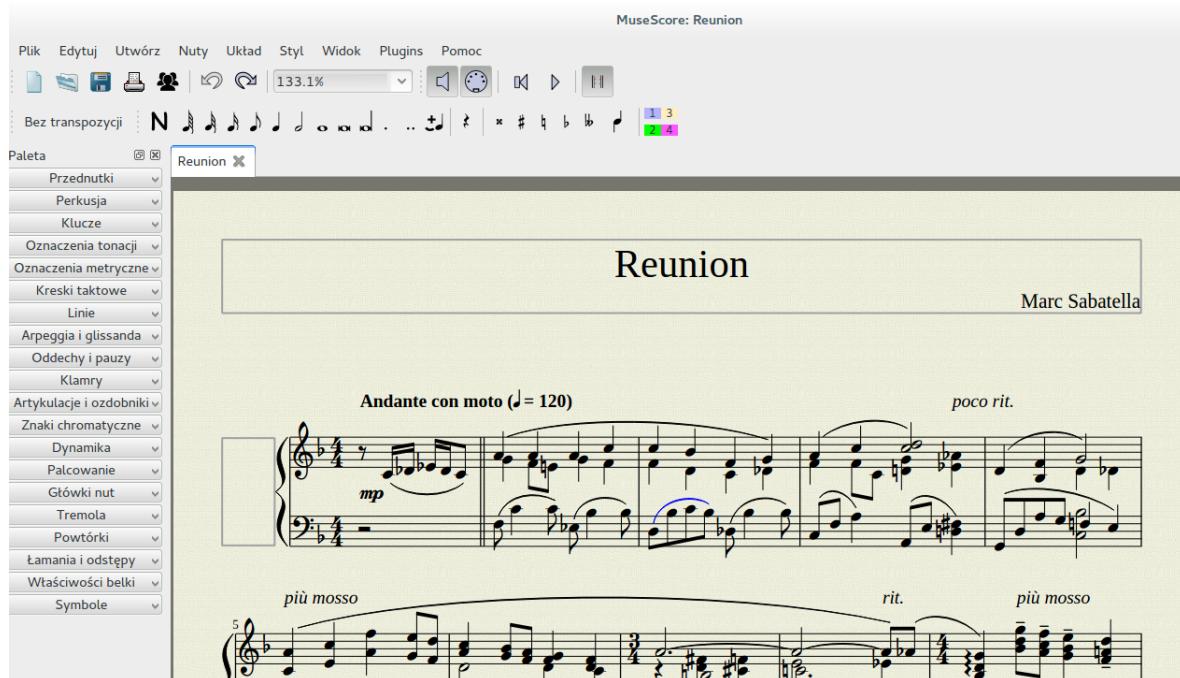
2.2 Popularne edytory notacji muzycznej

Edytory notacji muzycznej służą do tworzenia partytur. Podstawowa funkcjonalność edytora to tworzenie, edycja, drukowanie, eksport, import oraz odgrywanie partytur. Powszechnie dostępne metody wprowadzania danych to: mysz komputerowa, klawiatura MIDI, klawiatura komputerowa, wirtualne pianino. Najpopularniejsze formaty, do których eksportuje się partytury to PDF, MusicXML, MIDI, formaty audio (WAV, MP3, OGG), formaty graficzne (BMP, GIF, TIFF, JPEG, PNG, SVG). Import zwykle dostępny jest z formatów MIDI, MusicXML. Do innych popularnych funkcji należą: metronom, notacja dla perkusji, nazwy akordów, diagramy akordów, tabulatura gitarowa, edytowanie myszką, transponowanie, palety narzędzi, mikser głośności, centrum wsparcia technicznego.

2.2.1 Darmowe

2.2.1.1 MuseScore

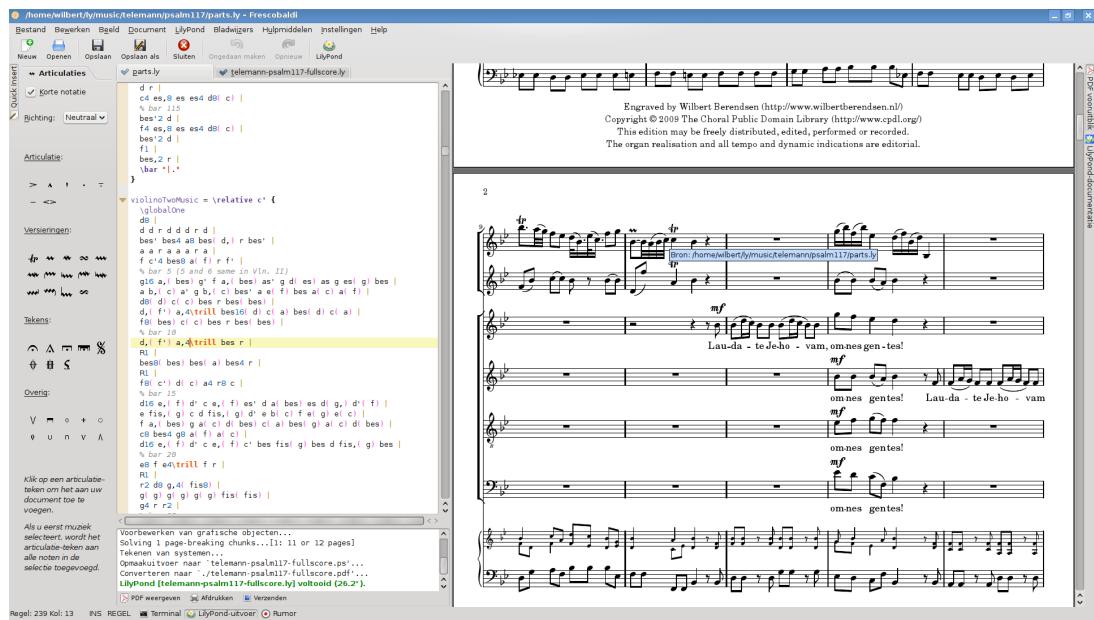
- **Rok powstania:** 2002,
- **Producent:** Werner Schweer i inni,
- **Stabilna wersja:** 1.3 / luty 2013,
- **URL:** <http://musescore.org/>,
- Multiplatformowy (Windows, Mac OS, Linux),
- Otwarty kod źródłowy,
- Platforma wymiany partytur MuseScore Connect [20],
- Wtyczki.



Rysunek 2.10: MuseScore

2.2.1.2 LilyPond

- **Rok powstania:** 1996,
- **Producent:** LilyPond development team,
- **Stabilna wersja:** 2.18.0 / grudzień 2013,
- **URL:** <http://lilypond.org/>,
- Multiplatformowy (Windows, Mac OS, Linux),
- Otwarty kod źródłowy,
- Język kompozycji.



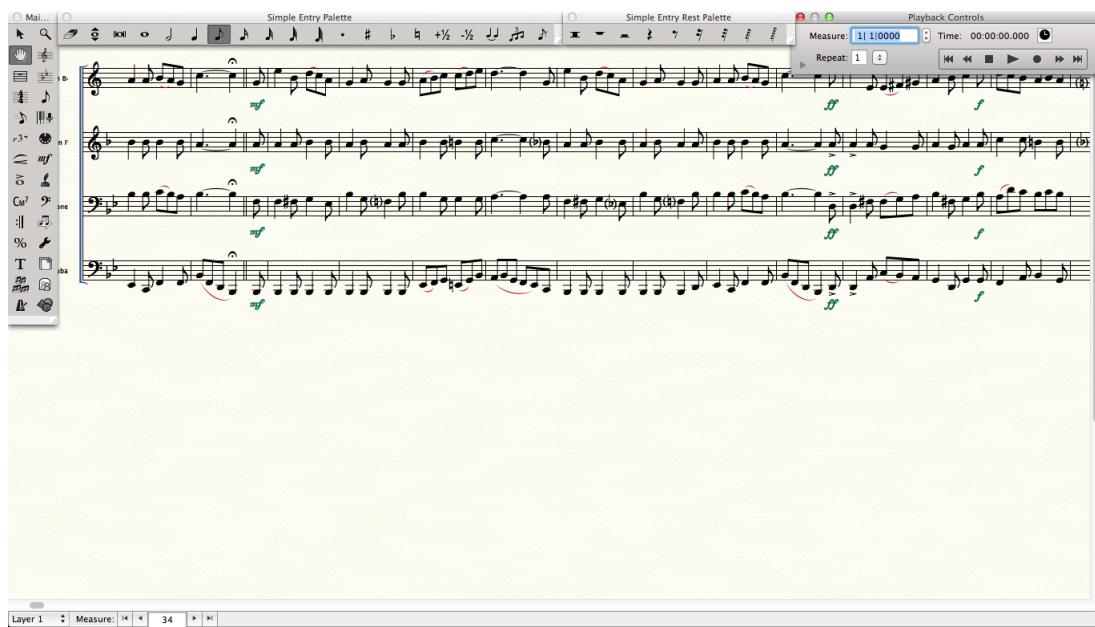
Rysunek 2.11: LilyPond (widok z GUI Frescobaldi)

Źródło: <http://www.lilypond.org/>

2.2.2 Płatne

2.2.2.1 Finale

- **Rok powstania:** 1988,
- **Producent:** MakeMusic,
- **Stabilna wersja:** 2014b / czerwiec 2014,
- **URL:** <http://www.finalemusic.com/>,
- Cena: 600 \$,
- Multiplatformowy (Windows, Mac OS),
- Biblioteka dźwięków Garritan (wysokiej jakości programowe instrumenty syntezowane),
- Nieograniczone cofanie zmian,
- Darmowa przeglądarka plików.

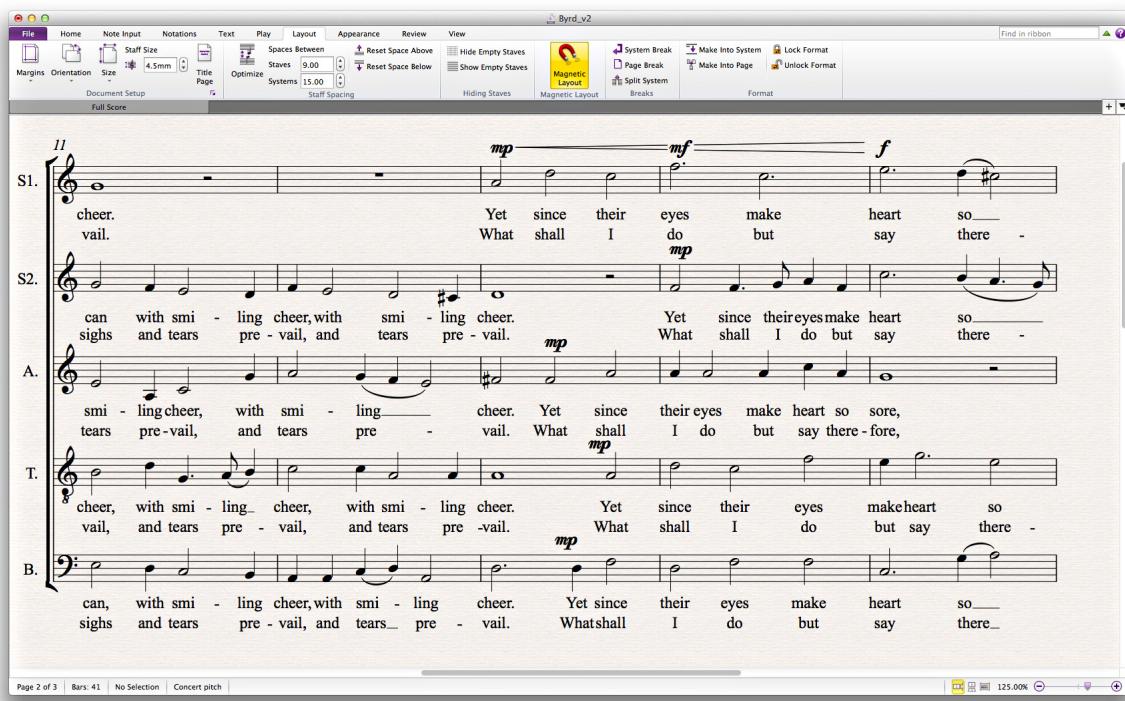


Rysunek 2.12: Finale

Źródło: <http://forum.makemusic.com/>

2.2.2.2 Sibelius

- **Rok powstania:** 1993,
- **Producent:** Sibelius Software,
- **Stabilna wersja:** 7.5 / luty 2014,
- **URL:** <http://www.sibelius.com/>,
- Cena: 600 \$,
- Multiplatformowy (Windows, Mac OS),
- Wersjonowanie partytur,
- Wirtualne pianino,
- PhotoScore (oprogramowanie OMR).



Rysunek 2.13: Sibelius

Źródło: <http://www.sibelius.com/>

2.2.3 Porównanie funkcjonalności

2.2.3.1 Obsługiwane formaty plików

Tablica 2.1: Zestawienie możliwości importu danych analizowanych programów do edycji notacji muzycznej.

Format\Program	MuseScore	LilyPond	Finale	Sibelius
Bagpipe Music Writer	tak			
BB	tak			
Capella	tak			
ENIGMA			tak	
Finale Auto Save			tak	
Finale Backup			tak	

Finale Legacy Notation			tak	
Finale Legacy Template			tak	
Finale Notation			tak	
Finale Performance Assessment			tak	
Finale Template			tak	
Lesson			tak	
MIDI	tak		tak	tak
Muse Data	tak			
Musescore	tak			
MusicXML	tak	tak	tak	tak
Overture	tak			
PhotoScore/AudioScore				tak
Sibelius				tak
Skompresowany Musescore	tak			
Skompresowany MusicXML	tak		tak	tak
SmartScore Lite Scan			tak	
TIFF			tak	

Tablica 2.2: Zestawienie możliwości **eksportu** danych analizowanych programów do edycji notacji muzycznej.

Format\Program	MuseScore	LilyPond	Finale	Sibelius
Avid Scorch				tak
BMP				tak
Encapsulated PostScript				tak
EPUB			tak	
Finale Legacy Notation			tak	
Finale Notation			tak	

Finale Template			tak	
Flac Audio	tak			
LilyPond	tak			
ManuScript				tak
MIDI	tak	tak	tak	tak
MP3			tak	
MuseScore	tak			
MusicXML	tak		tak	tak
Ogg Vorbis Audio	tak			
PDF	tak	tak	tak	tak
PNG	tak	tak		tak
PostScript	tak	tak		
Scorch Web Page				tak
Sibelius				tak
Sibelius Legacy				tak
Skompresowany MuseScore	tak			
Skompresowany MusicXML	tak		tak	tak
SmartMusic			tak	
SVG	tak			tak
TIFF				tak
Wave Audio	tak		tak	tak
Windows Media Video				tak

2.2.3.2 Podsumowanie

Pomimo faktu, że oprogramowanie Finale jest płatne, dostępna jest darmowa przeglądarka dla plików w nim stworzonych. Finale w przeciwieństwie do Sibelius

posiada opcję nieograniczonego cofania wprowadzonych zmian. Sibelius natomiast posiada opcje wersjonowania partytur, wirtualną klawiaturę pianina oraz tak zwany tryb kontroli klasy, przydatny dla nauczycieli pracujących w sieci komputerowej razem z uczniami. MuseScore w przeciwieństwie do Finale oraz Sibelius posiada system wtyczek pisanych w QtScript, języku podobnym do JavaScript. Inną opcją występującą tylko w MuseScore jest MuseScore Connect pozwalające na interakcję z internetową społecznością tworzącą partytury, publikowanie własnych partytur w Internecie. Z pewnością LilyPond odróżnia się od innych prezentowanych tutaj rozwiązań ze względu na bycie swoistym językiem tekstowym do opisu partytur. LilyPond posiada m.in. moduł odpowiadający za automatyczne skuteczne formatowanie, przeciwieństwem tego rozwiązania w programach WYSIWYG (ang. What You See Is What You Get) jest możliwość samodzielnego przenoszenia elementów notacji. We wszystkich analizowanych programach dostępny jest import MusicXML. Wszystkie analizowane programy obsługują eksport do MIDI, PDF. Pozostałe popularne formaty eksportu to MusicXML, PNG, Skompresowany MusicXML, Wave Audio (wszystkie obecne w trzech z czteru analizowanych programów).

2.3 Popularne oprogramowanie do optycznego rozpoznawania muzyki

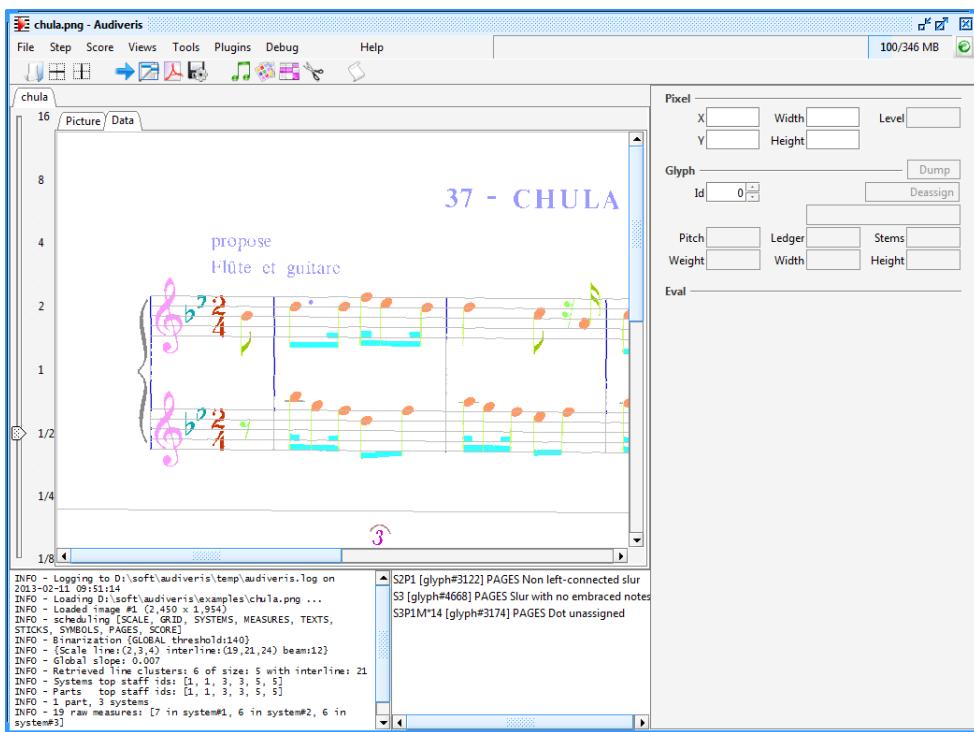
Optyczne rozpoznawanie muzyki (ang. OMR od Optical Music Recognition) inaczej nazywane Music OCR (Music Optical Character Recognition) opiera się na zastosowaniu optycznego rozpoznawania znaków do konwersji obrazów partytur do edytowalnych formatów danych.

2.3.1 Darmowe

2.3.1.1 Audiveris

- **Rok powstania:** 2000,
- **Producent:** Hervé Bitteur,

- Stabilna wersja: 4.3 / listopad 2013,
- URL: <https://audiveris.kenai.com/>,
- Multiplatformowy (Windows, Linux),
- Napisany w Java,
- Obsługa tylko drukowanych partytur.



Rysunek 2.14: Audiveris

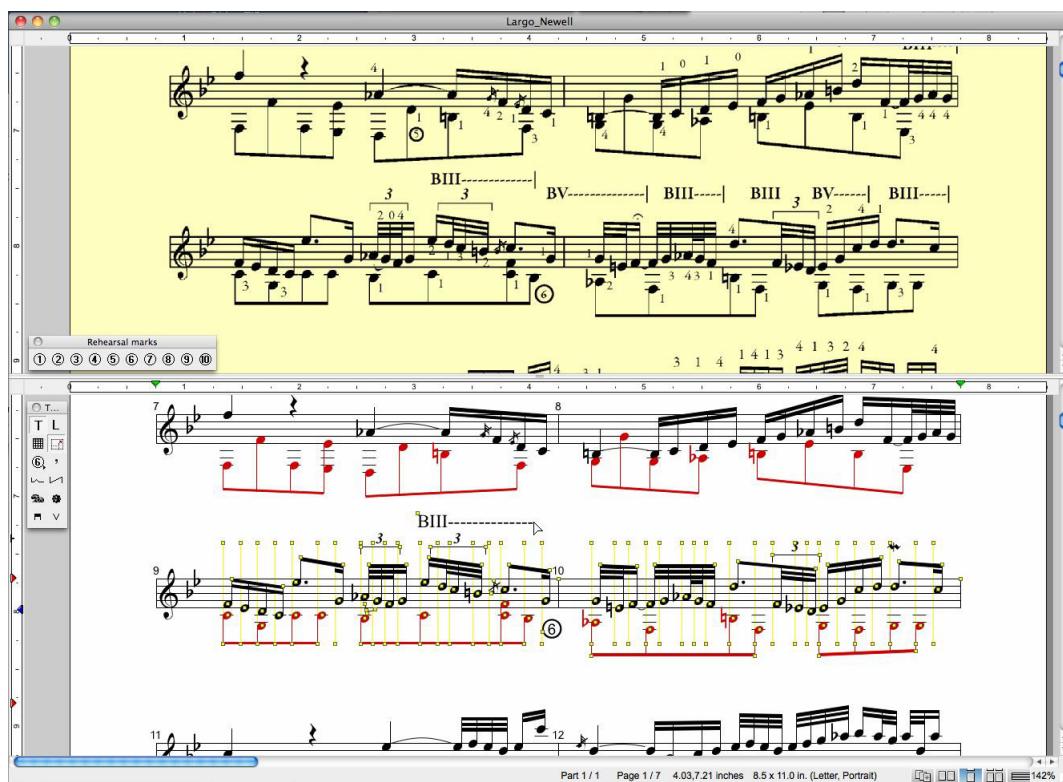
Źródło: <https://audiveris.kenai.com/>

2.3.2 Płatne

2.3.2.1 SmartScore

- Rok powstania: 1991,
- Producent: Musitek Corporation,
- Stabilna wersja: 10.5.3 / kwiecień 2013,
- URL: <https://www.musitek.com/smartscore-pro.html>,

- Cena: 399 \$,
- Multiplatformowy (Windows, Mac OS),
- Wspiera partytury drukowane oraz częściowo partytury pisane ręcznie,
- Rozpoznaje symbole i diagramy akordów,
- Rozpoznaje tabulatury, perkusje,
- Biblioteka dźwięków Garritan (wysokiej jakości programowe instrumenty syntezowane).



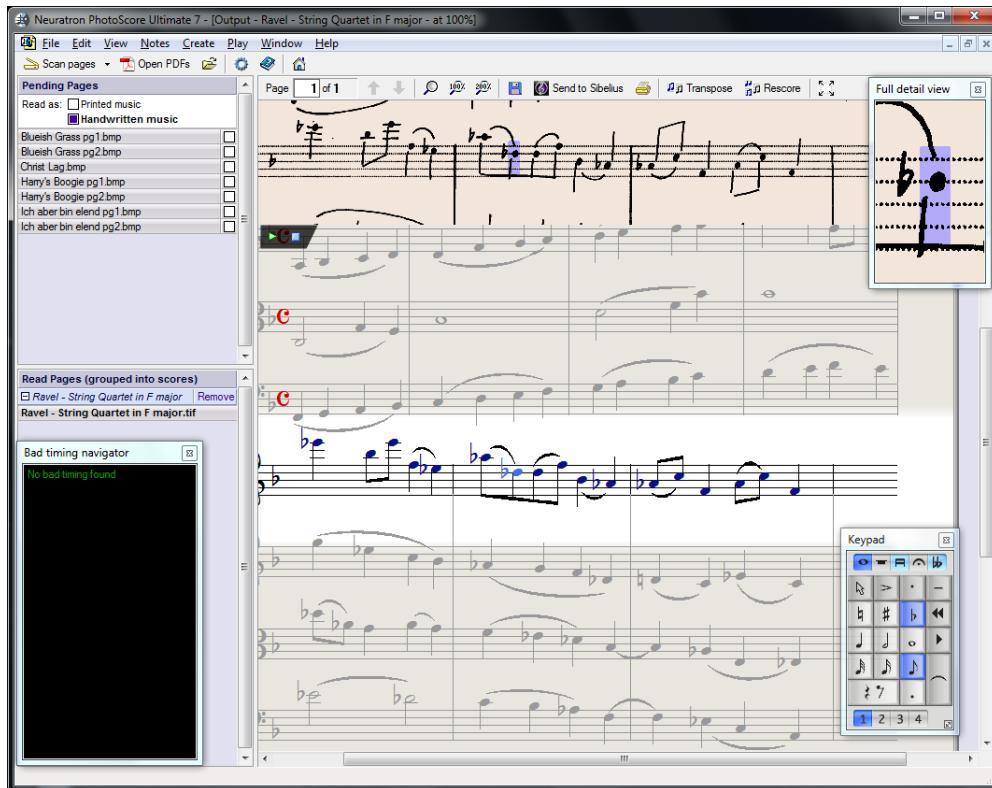
Rysunek 2.15: SmartScore

Źródło: <http://www.musitek.com/>

2.3.2.2 Photoscore Ultimate

- **Rok powstania:** 2007,
- **Producent:** Neuratron,
- **Stabilna wersja:** 7 / 2011,

- URL: <http://www.neuratron.com/photoscore.htm>,
- Cena: 250 \$,
- Multiplatformowy (Windows, Mac),
- Wspiera partytury drukowane i pisane ręcznie.



Rysunek 2.16: Photoscore

Źródło: <http://www.neuratron.com/>

2.3.2.3 NotateMe

- Rok powstania: 2013,
- Producent: Neuratron,
- Stabilna wersja: 2.5.0.3 / czerwiec 2014,
- URL: <http://www.neuratron.com/notateme.html>,
- Cena: 139 zł,
- Multiplatformowy (iOS, Android),

- Aplikacja mobilna,
- Wprowadzanie notacji online,
- Darmowa wersja NotateMe Now dla pojedynczej pięciolinii.



Rysunek 2.17: NotateMe

Źródło: <http://www.neuratron.com/>

2.3.2.4 Inne warte uwagi oprogramowanie

- capella-scan — <http://www.capella.de/us/index.cfm/products/capella-scan/info-capella-scan/>,
- SharpEye — <http://www.visiv.co.uk/>,
- ThinkMusic — <http://thinkmusictech.com/> (produkt nieopublikowany).

2.3.3 Porównanie funkcjonalności

2.3.3.1 Obsługiwane formaty plików

Tablica 2.3: Zestawienie możliwości **importu** danych analizowanych programów do optycznego rozpoznawania muzyki.

Format\Program	Audiveris	SmartScore	Photoscore Ultimate	NotateMe
BMP	tak	tak	tak	
GIF	tak			
JPG	tak			
MIDI		tak		
Music XML		tak		tak
NIFF		tak		
NotateMe				tak
PDF	tak	tak	tak	
PDF	tak			
PhotoScore			tak	
PNG	tak			
SmartScore		tak		
TIF	tak	tak		
TIFF	tak	tak	tak	

Tablica 2.4: Zestawienie możliwości **eksportu** danych analizowanych programów do optycznego rozpoznawania muzyki.

Format\Program	Audiveris	SmartScore	Photoscore Ultimate	NotateMe
AIFF		tak		
Audiveris XML Script	tak			
BMP		tak		
MIDI			tak	tak
Music XML	tak		tak	tak
NIFF			tak	
NotateMe				tak

PDF		tak		
PDF				tak
PhotoScore			tak	
TIF		tak		
Wave Audio			tak	

2.3.3.2 Podsumowanie

Przede wszystkim należy odróżnić programy skanujące muzykę zapisaną/zdigitalizowaną od programów przetwarzających dane online (na bieżąco). Do pierwszej grupy zaliczają się: Audiveris, SmartScore, PhotoScore, capella-scan; w przeciwieństwie do NotateMe oraz produktu w przygotowaniu ThinkMusic. Z wszystkich wymienionych programów do rozpoznawania notacji pisanej ręcznie nadają się tylko SmartScore, PhotoScore, NotateMe, pozostałe programy czytają tylko partytury drukowane. Najwięcej programów obsługuje import z formatów PDF oraz TIFF. Popularnym formatem do eksportu danych okazuje się MusicXML, dostępny w trzech z czterech analizowanych programów.

Rozdział 3

Projekt programu

Program nazwano DrawNotes ze względu na oferowaną funkcjonalność, tj. możliwość wprowadzania nut do programu do ich edycji poprzez rysowanie.

3.1 Badanie znajomości programów do edycji notacji muzycznej

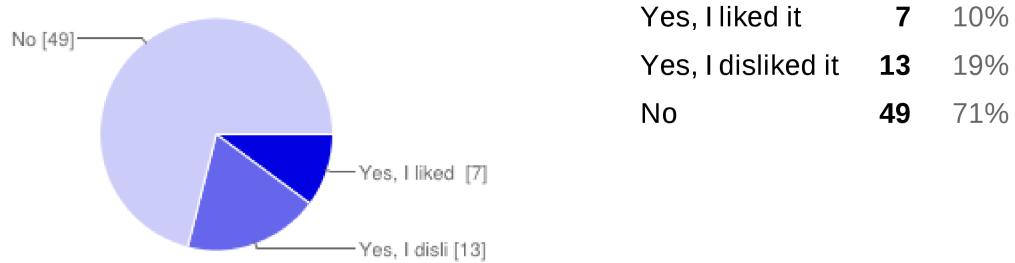
W celu zdobycia wiedzy o problemie przeprowadziłem badanie znajomości programów do edycji notacji muzycznej na grupie 70 osób. W skład tej grupy wchodziły osoby mi znajome oraz użytkownicy IRC kanałów #musescore, #lilypond, #music, #music-theory z serwera freenode.net (<http://freenode.net>). Ankietę przygotowałem w języku angielskim. Zadałem następujące pytania:

1. „Have you tried any handwritten music recognition applications?” — Czy próbowałeś jakiekolwiek aplikacje rozpoznająceisaną ręcznie muzykę?
2. „Would you want to try out new handwritten musical notation editor?” — Czy chciałbyś wypróbować nowy edytor muzyki pisanej ręcznie?
3. „Would you pay for handwritten musical notation editor?” — Czy zapłaciłbyś za edytor notacji muzycznej pisanej ręcznie?
4. „Are you happy with music notation input possibilities?” — Czy jesteś zadowolony z możliwości wprowadzania notacji muzycznej?

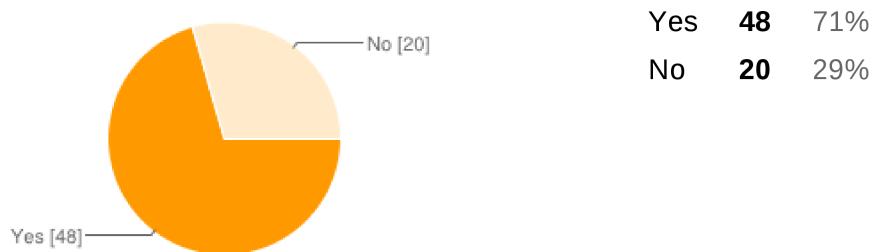
5. „Which methods for music notation input do you use?” — Jakich metod do wprowadzania notacji muzycznej używasz? Odpowiedzi w formie wielokrotnego wyboru:
 - „computer keyboard shortcuts” — skróty klawiszowe,
 - „mouse” — myszka,
 - „composition language (e.g. LilyPond)” — język kompozycji np. LilyPond,
 - „MIDI keyboard” — klawiatura MIDI,
 - „handwritten” — pisanie ręcznie,
 - „other” — inne.
6. „Share your thoughts on handwritten music notation input methods.” — Podejrzewaj się przemyśleniami na temat metod wprowadzania notacji muzycznej poprzez pisanie ręczne.

Na rysunkach 3.1 oraz 3.2 znajdują się wyniki ankiety.

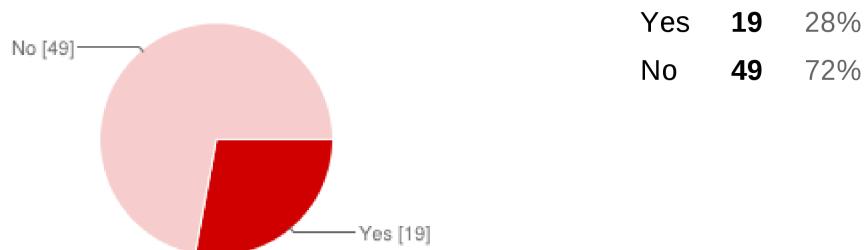
Have you tried any handwritten music recognition applications?



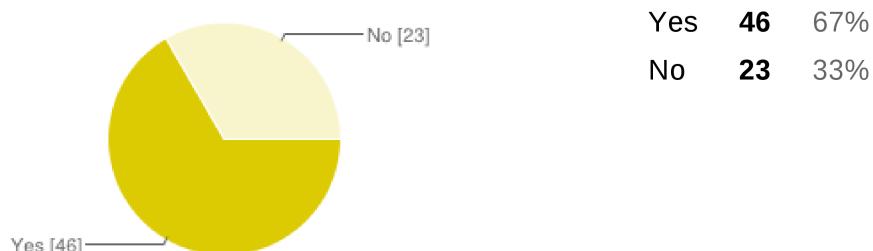
Would you want to try out new handwritten musical notation editor?



Would you pay for handwritten musical notation editor?

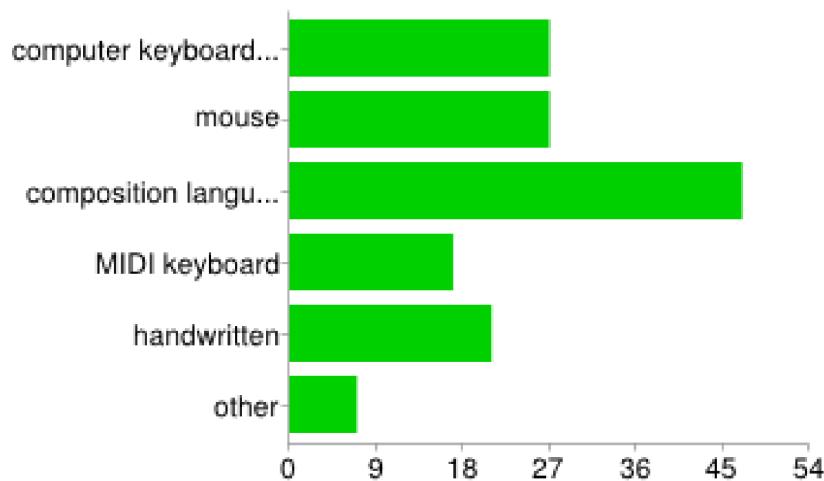


Are you happy with music notation input possibilities?



Rysunek 3.1: Wyniki badania dla pytań 1-4.

Which methods for music notation input do you use?



computer keyboard shortcuts	27	18%
mouse	27	18%
composition language (e.g. LilyPond)	47	32%
MIDI keyboard	17	12%
handwritten	21	14%
other	7	5%

Rysunek 3.2: Wyniki badania dla pytania 5.

W odpowiedzi na pytanie otwarte pojawiły się następujące głosy:

- „Chciałbym mieć możliwość wprowadzania notacji muzycznej z poziomu tabletu.”
- „Skorzystałbym z opcji wprowadzania notacji muzycznej przez pisanie ręczne, gdyby zaoszczędzało mi to czasu.”
- „Wszystkie transkrypcje piszę w LilyPond i jestem z tego sposobu zadowolony.”
- „Ze względu na niedokładność pisania ręcznego używam do wpisywania znaków klawiatury.”
- „Jestem bardzo sceptycznie nastawiony do pomysłu. Podczas korzystania

z NotateMe czuję się jakbym cały czas walczył z programem, by ten zrozumiał prostą sekwencję nut.”

- „Edytor notacji muzycznej pisanej ręcznie byłby dla mnie interesujący, gdyby był:
 - a) otwartoźródłowy,
 - b) eksportował do LilyPond lub co najmniej do MusicXML,
 - c) pisanie byłoby zauważalnie szybsze niż w LilyPond.”

3.1.1 Wnioski

Poniżej wnioski z badania.

- Większość (71% przebadanych osób) nie miała styczności z aplikacjami pozwalającymi na rozpoznawanie notacji muzycznej pisanej ręcznie. Dwie trzecie osób mających styczność z takimi aplikacjami negatywnie ocenia swoje doświadczenia z nimi.
- Większość (71% przebadanych osób) chciałaby wypróbować nowy edytor notacji muzycznej pisanej ręcznie.
- Mniejszość (28% przebadanych osób) zapłaciłaby za edytor notacji muzycznej pisanej ręcznie.
- Mniejszość (33% przebadanych osób) nie jest zadowolona z istniejących rozwiązań do wprowadzania notacji muzycznej.
- Spośród metod wprowadzania notacji muzycznej przeważają języki kompozycji (np. LilyPond), następnie popularnością cieszą się klawiatura i myszka.
- W odpowiedzi na pytanie otwarte pojawiły się sceptyczne głosy, które jednoznacznie wskazują, że proponowane rozwiązanie musi posiadać szereg zalet, by zostało zaakceptowane przez użytkowników, takie jak: szybkość i dokładność wprowadzania elementów, intuicyjność, darmowość.

3.2 Funkcjonalność programu i wybrane oprogramowanie

Program ma za zadanie realizować następujące funkcje:

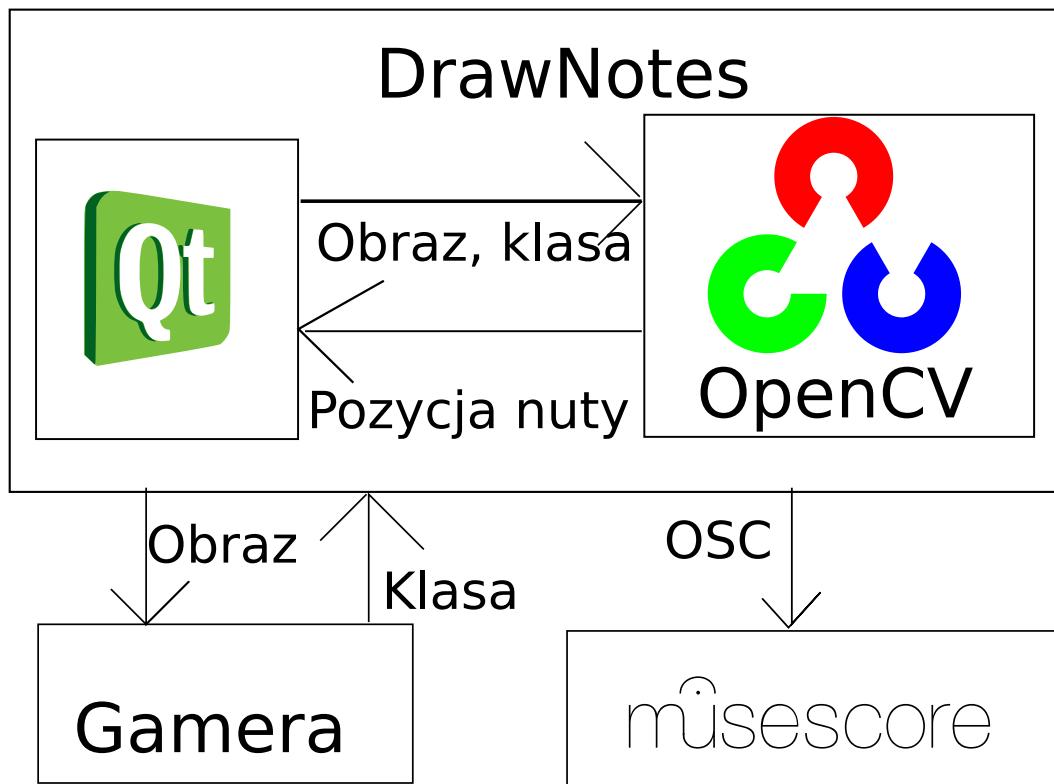
- Rozpoznaje nuty (cała nuta, półnuta, ćwierćnuta, ósemka, szesnastka, trzydziestka dwójka),
- Rozpoznaje wysokość nuty,
- Komunikuje się za pomocą protokołu OSC z MuseScore.

Do klasyfikacji nut zaplanowano użycie oprogramowania do budowania aplikacji analizujących dokumenty Gamera [24].

MuseScore to darmowe oprogramowanie do kompozycji i notacji muzycznej o otwartym kodzie źródłowym, aktywnie rozwijane. Po konsultacjach z programistami MuseScore zdecydowano wykorzystać OSC (Open Sound Control) do komunikacji z MuseScore w celu edycji partytury. OSC to protokół komunikacyjny umożliwiający wymianę danych w czasie rzeczywistym pomiędzy aplikacjami, instrumentami muzycznymi oraz innymi urządzeniami.

3.3 Architektura

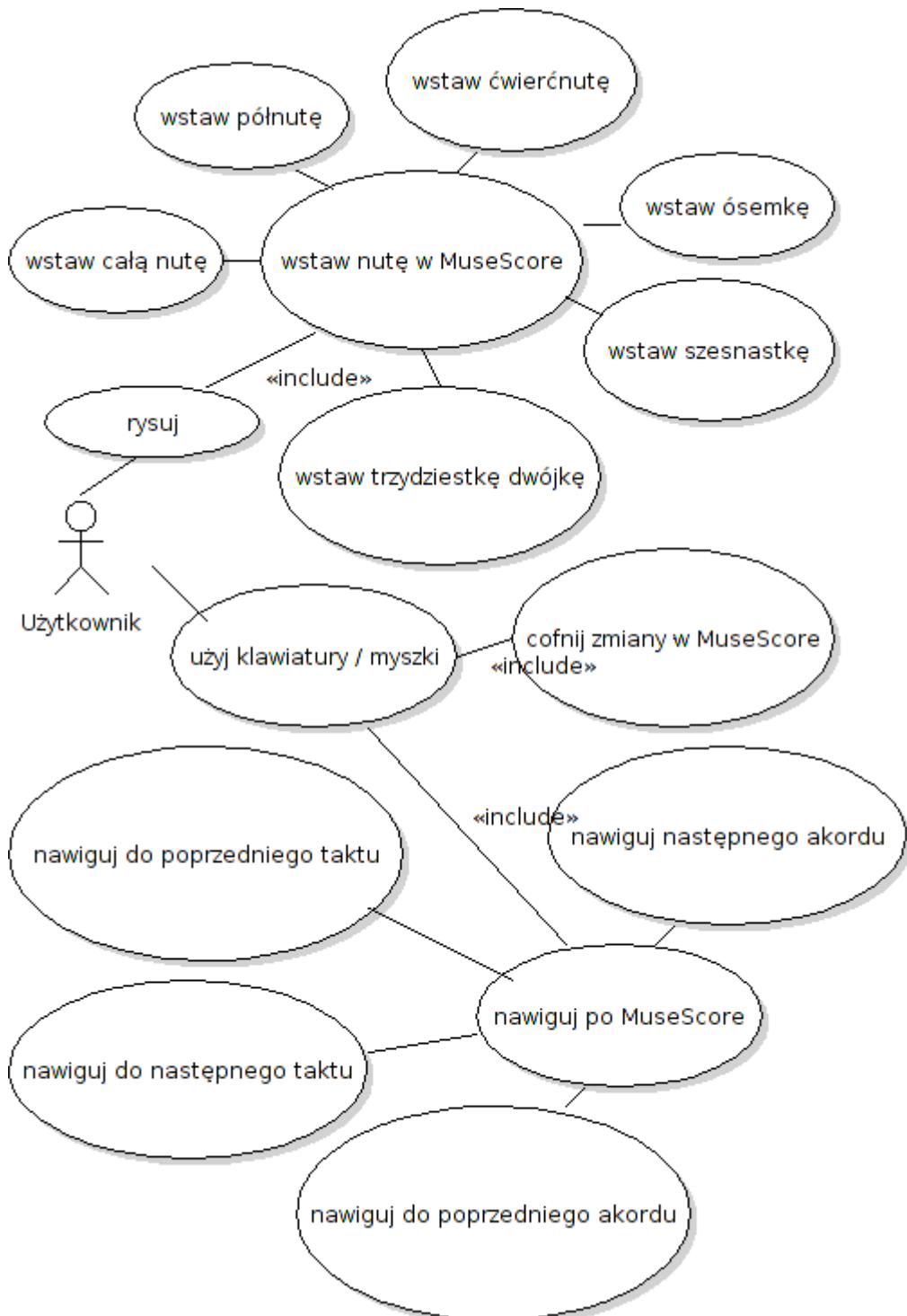
Poniżej przedstawiam architekturę programu DrawNotes.



Rysunek 3.3: Architektura programu DrawNotes

3.4 Diagram przypadków użycia

Diagram przypadków użycia opisuje podstawowe czynności jakie może wykonać użytkownik z pomocą programu DrawNotes.



Rysunek 3.4: Diagram przypadków użycia programu DrawNotes

Rozdział 4

Implementacja programu

4.1 Proces rozpoznawania symboli

Do realizacji programu DrawNotes użyto następujących pakietów oprogramowania Qt [19], Gamera [24] oraz OpenCV [17].

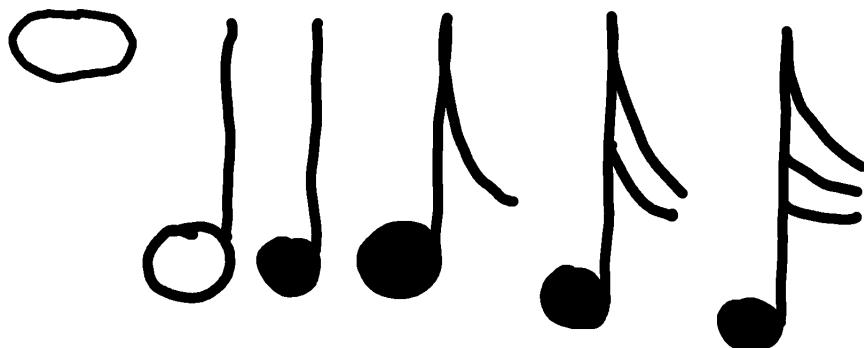
Rozpoznanie nuty odbywa się z wykorzystaniem projektu Gamera. Klasyfikacja odbywa się z pomocą algorytmu k najbliższych sąsiadów oraz następujących cech obrazu:

- moments — momenty obrazu. Ich definicję można znaleźć m.in. w książce „Digital Image Processing” [5].
- nholes — oblicza dla każdego wiersza lub kolumny średnią liczbę białych przebiegów niedotykających brzegu. Z tych liczb, zwracana jest średnia dla wierszy i kolumn.
- nholes_extended — dzieli obraz na cztery paski i wtedy wykonuje analizę nholes na każdy z pasków. Jest to wykonywane wertykalnie i następnie horyzontalnie, co w rezultacie daje osiem wartości cech.
- volume — procent czarnych pikseli wewnątrz prostokątnej obwiedni obrazu.
- zernike_moments — oblicza wartości absolutne znormalizowanych momentów Zernike, które są niezmienne dla operacji skalowania, rotacji i odbicia. Ich definicję można znaleźć m.in. w książce „Image Analysis by Moments” [7].

Powyższe opisy powstały głównie poprzez tłumaczenie opisów cech znajdujących się na stronie internetowej projektu Gamera [6]. Użyty został algorytm k najbliższych sąsiadów, gdzie $k = 1$, ponieważ w dokumentacji Gamera czytamy „During classification, you can choose two parameters: the parameter k in the kNN classifier, and the set of features. The choice of k depends on the minimum number of training samples you have for each class. [...] in most practical cases this boils down to k = 1, unless you have an abundant amount of training data” [1]. Na wejściu algorytmu k-nn mamy: zbiór obiektów na podstawie, których przebiegać będzie klasyfikacja (dane treningowe), obiekt do sklasyfikowania, oraz parametr k. Na wyjściu otrzymujemy decyzję, do której klasy został zaklasyfikowany obiekt. Algorytm składa się z dwóch kroków:

1. Poszukaj obiektów najbliższych w stosunku do obiektu klasyfikowanego na podstawie cech obiektów,
2. Określ klasę obiektu klasyfikowanego na podstawie obiektów najbliższych.

Dla każdego symbolu przygotowano zestaw jego 25 wzorcowych obrazów. Poniżej przykładowe symbole.



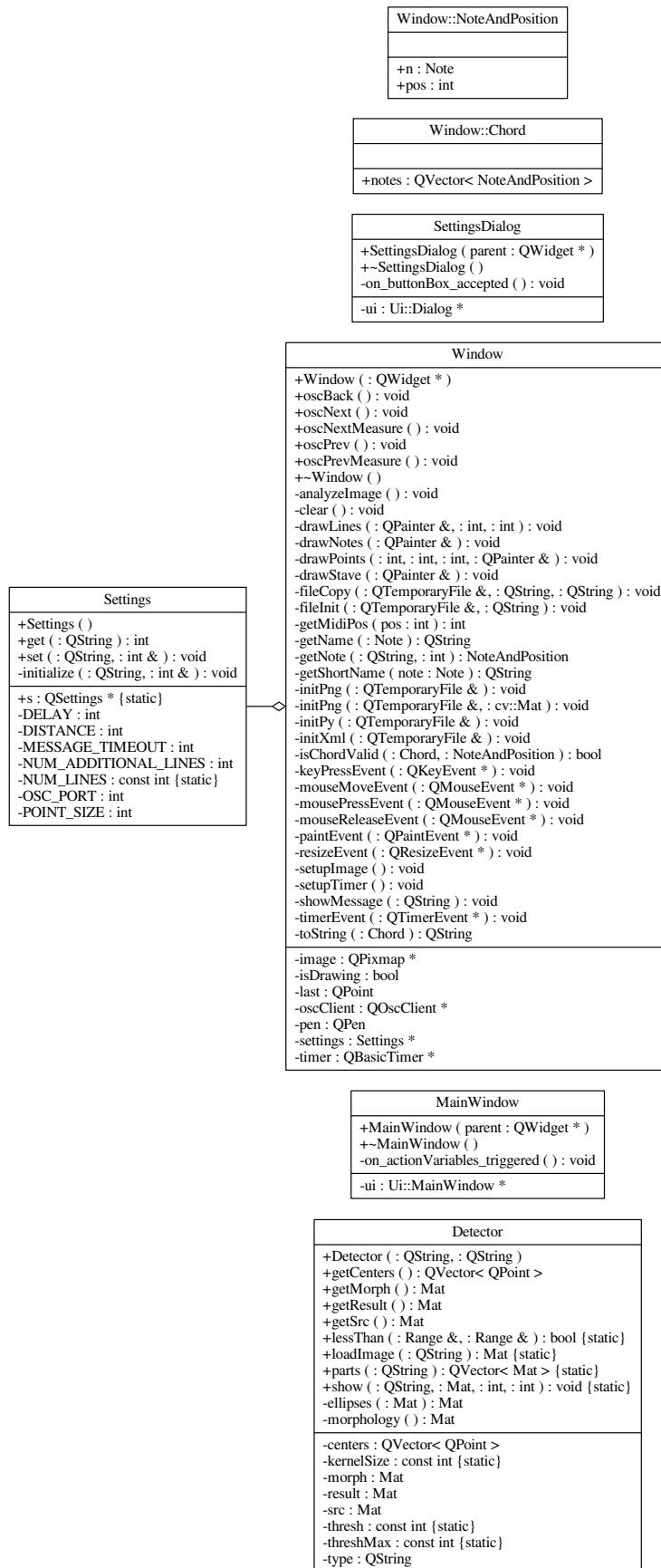
Rysunek 4.1: Przykładowe ręcznie narysowane symbole nut

Rozpoznanie wysokości nuty odbywa się trójetapowo:

1. Obraz jest segmentowany na spójne części,
2. W przypadku całej nuty oraz półnuty następuje zamalowanie dziur z wykorzystaniem: algorytmu Canny [3] do znajdowania konturów obrazu; momentów Hu [8] do obliczenia centrum dziury; algorytmu floodFill do zamalowania

dziury [11]. Następnie dla wszystkich nut z wyjątkiem całej nuty następuje morfologiczne zamykanie w celu usunięcia lasek nut [12],

3. Ostatnim krokiem jest wykonanie algorytmu fitEllipse [4] na konturach zmodyfikowanego obrazu w celu poznania współrzędnych środków główek nut.



Rysunek 4.2: Diagram klas programu DrawNotes

4.2 MuseScore

Konieczna była implementacja obsługi następujących akcji wywoływanych za pomocą protokołu OSC:

- next-chord, prev-chord — przejdź do następnego, poprzedniego akordu,
- next-measure, prev-measure — przejdź do następnego, poprzedniego taktu,
- backspace — obsługa klawisza backspace — cofanie zmian,
- pad-note-X — wybierz czas trwania wprowadzanych nut (gdzie X przyjmuje wartości: 1, 2, 4, 8, 16, 32),
- add-pitch — dodaj w bieżącym miejscu nutę o zadanej wysokości i poprzednio określonym czasie trwania (patrz wyżej).

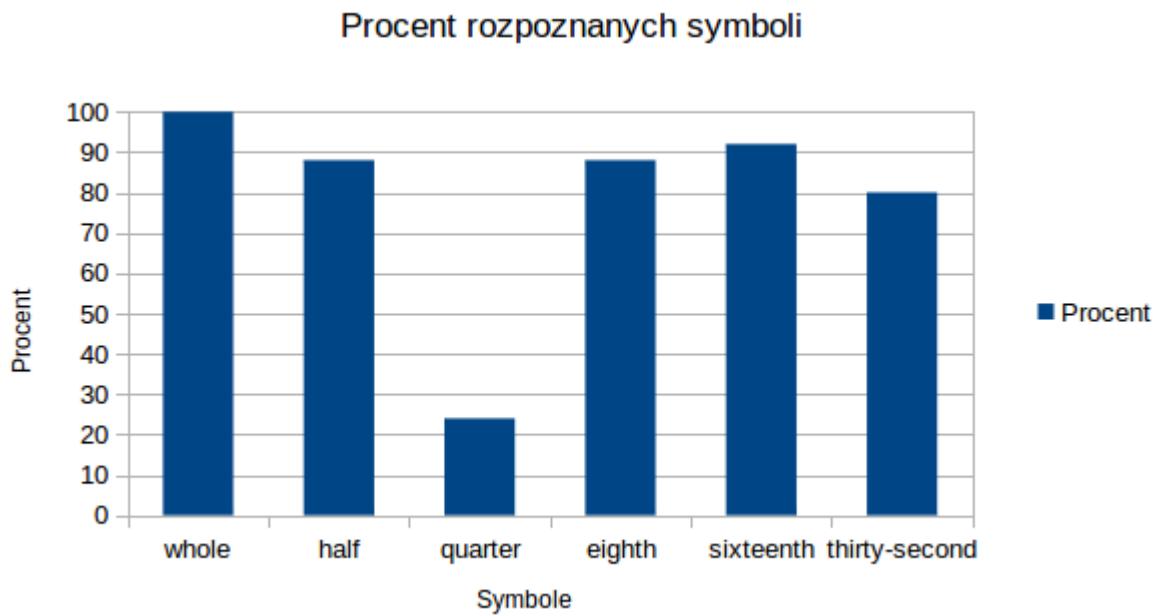
4.3 Testy

4.3.1 Klasyfikacja

Przygotowałem skrypt testowy, który działa następująco:

1. Pobiera argumenty: testowany katalog z plikami PNG, położenie skryptu klasyfikatora, położenie klasyfikatora XML,
2. Dla każdego pliku w testowanym katalogu klasyfikuje go i sprawdza, czy przypnana klasa zgadza się z klasą zadeklarowaną w nazwie pliku (format «klasa><nr>.png»),
3. Wyświetla liczbę pozytywnie sklasyfikowanych obrazów.

W moich testach 100% symboli testowych dużej rozdzielczości było rozpoznawanych. Natomiast symbole w małej rozdzielczości były rozpoznawane w około 80% przypadków. Rozmiar danych testowych wynosił 5 dla każdego symbolu dużej rozdzielczości oraz 25 dla każdego symbolu małej rozdzielczości. Symbole testowe wyglądają podobnie do tych pokazanych wcześniej na rysunku 4.1.

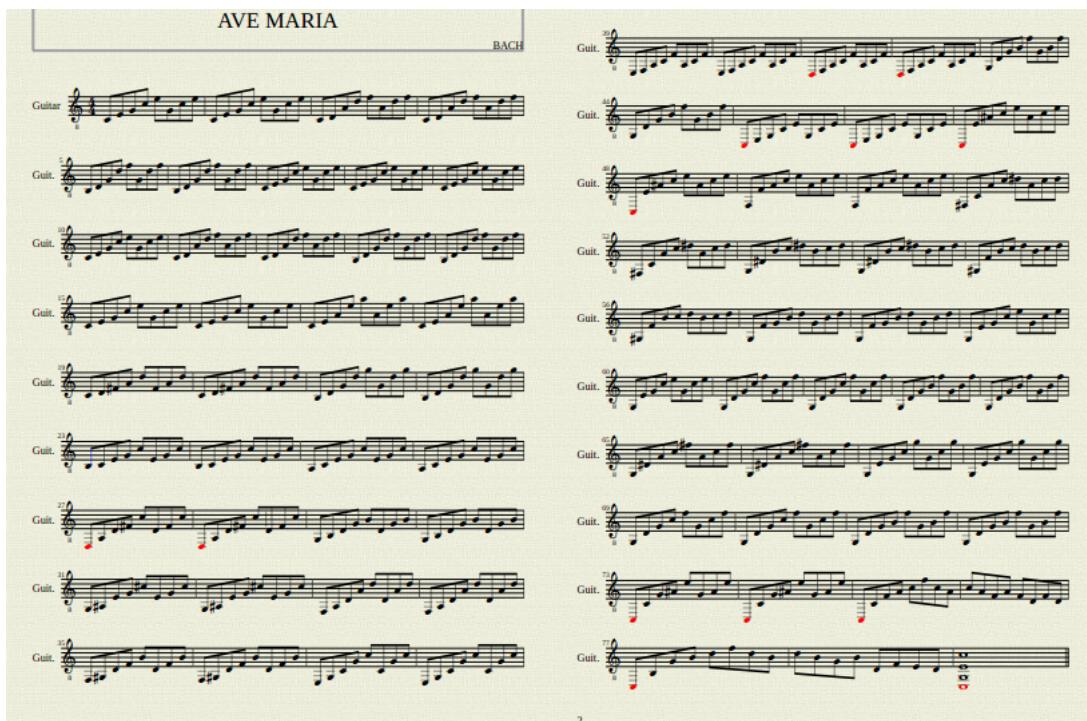


Rysunek 4.3: Wynik eksperymentalnej klasyfikacji odręcznie narysowanych symboli testowych

W teście źle wypada symbol „quarter” czyli ćwierćnuta. Dzieje się tak prawdopodobnie ze względu na źle narysowane przykłady uczące, że zbyt długą laską. Ponadto symbol ten był często błędnie rozpoznawany jako „eighth”, czyli ósemka.

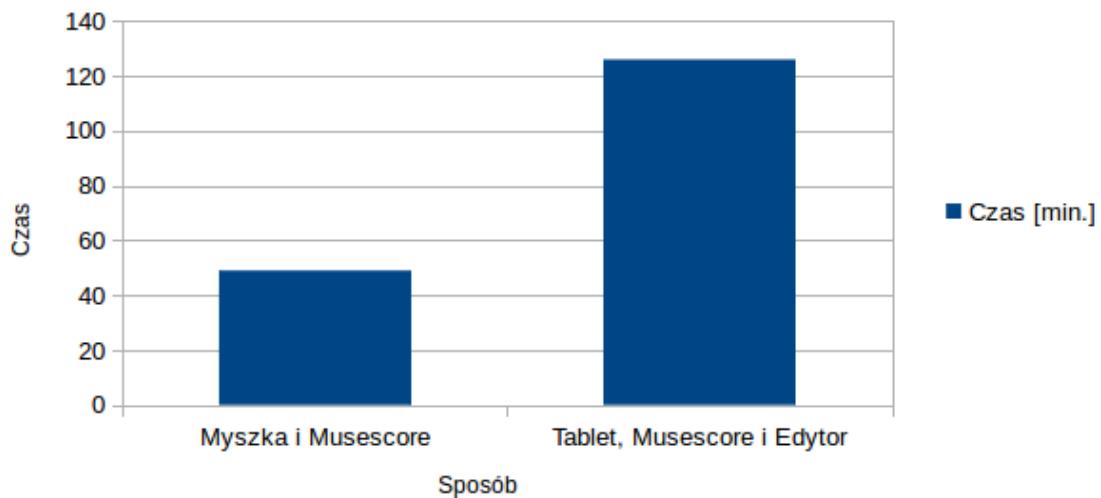
4.3.2 Próba czasowa

Wykonałem próbę czasową dwóch sposobów wprowadzania notacji muzycznej — wprowadzania za pomocą myszki oraz poprzez zbudowany program DrawNotes (pisanie ręczne). Do testu użyłem znanego utworu „Ave Maria” J.S. Bacha 4.4.



Rysunek 4.4: Przepisywany utwór — Ave Maria J.S. Bacha.

Pomiar czasu przepisywania "Ave Maria" J.S. Bacha



Rysunek 4.5: Próba czasowa

Z rysunku 4.5 można odczytać, że sposób wprowadzania notacji muzycznej przez pisanie ręczne jest około 2,5 raza wolniejszy (49 minut versus 126 minut).

Rozdział 5

Podsumowanie i wnioski

Cel pracy został osiągnięty. W szczególności udało się: dobrać i zintegrować narzędzia do rozpoznawania i cyfryzacji ręcznie sporzązonej notacji muzycznej w postaci programu DrawNotes współpracującego z MuseScore; przygotować zestawy danych treningowych (25 obrazów dla każdego symbolu) i testowych (30 obrazów dla każdego symbolu: 25 małej rozdzielczości i 5 dużej rozdzielczości); przeprowadzić i udokumentować testy (jakości klasyfikacji oraz szybkości przepisywania utworu); zbadać ankietowo znajomość narzędzi notacji muzycznej wśród użytkowników MuseScore i LilyPond.

Podczas testów okazało się jednak, że dokładność rozpoznawania symboli w małej rozdzielczości różni się od dokładności dla symboli z dużym DPI (ang. dots per inch), co wpływa na szybkość interakcji, ze względu na błędy rozpoznawania symboli. Dla symboli w małej rozdzielczości uzyskano dokładność rozpoznawania symboli wynoszącą około 80%. Przeprowadzone eksperymenty pokazały, że praca edycyjna z użyciem opracowanego narzędzia jest 2,5 raza wolniejsza niż tradycyjne wprowadzanie nut myszką. Z tego względu ocena metody jest negatywna.

Osiągniętym celem dydaktycznym, który jest efektem ubocznym pracy nad programem było poznanie klasyfikatorów Gamera, pakietów oprogramowania OpenCV oraz Qt. Jako że poruszany problem jest dość rozległy [23], zrealizowane oprogramowanie implementuje tylko podstawową funkcjonalność tj. wstawianie nut o różnym czasie trwania na pięciolinii do programu MuseScore za pośrednictwem protokołu OSC. W szczególności program nie rozpoznaje łączonych nut, akordów pisanych

w skróconej formie. Pomimo prostoty zrealizowanego oprogramowania pozwala ono na ocenę zaproponowanej metody wprowadzania notacji muzycznej. W porównaniu do pisania nut na papierze część użytkowników może poczuć się zniechęconych do korzystania z tego typu edytorów ze względu na wolne tempo tworzenia partytur. Rekompensatą za wolniejsze tempo pracy jest elektroniczny zapis partytury i możliwość błyskawicznego jej powielenia i eksportu do wielu popularnych formatów danych (pliki dźwiękowe, obrazy, PDF, MusicXML). Kluczem do zwiększenia tempa pracy z elektroniczną partyturą może być dopracowanie mechanizmu rozpoznawania symboli tak, by analizował on wprowadzane dane na bieżąco. Tego typu rozwiązania mogłyby zostać wprowadzone w ewentualnych kolejnych wersjach opracowanego programu.

Bibliografia

- [1] *A Tutorial Introduction to the Gamera Framework*. URL: <http://gamera.informatik.hsnr.de/docs/gamera-tutorial.pdf> (Odwiedzone 28.08.2014).
- [2] *Braille music summary by Bhugh at English Wikipedia*. URL: http://en.wikipedia.org/wiki/Braille_music#/mediaviewer/File:Braillemusicsummary.gif (Odwiedzone 25.08.2014).
- [3] *Canny Edge Detector*. URL: http://docs.opencv.org/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.html (Odwiedzone 18.03.2014).
- [4] *Creating Bounding rotated boxes and ellipses for contours*. URL: http://docs.opencv.org/doc/tutorials/imgproc/shapedescriptors/bounding_rotated_ellipses/bounding_rotated_ellipses.html (Odwiedzone 18.03.2014).
- [5] R.C. Gonzalez i R.E. Woods. *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006. ISBN: 013168728X.
- [6] *Features*. URL: <http://gamera.sourceforge.net/doc/html/features.html#nholes> (Odwiedzone 18.03.2014).
- [7] M. Pawlak. *Image Analysis by Moments: Reconstruction and Computational Aspects*. Wrocław: Oficyna wydawnicza Politechniki Wrocławskiej, 2006.
- [8] *Image Moments*. URL: <http://docs.opencv.org/doc/tutorials/imgproc/shapedescriptors/moments/moments.html> (Odwiedzone 18.03.2014).
- [9] S. Śledziński i A. Chodkowski. *Mała Encyklopedia Muzyki*. Państwowe Wydawnictwo Naukowe, 1968.

- [10] J.W. Reiss. *Mała historia muzyki*. Polskie Wydawnictwo Muzyczne, 1987.
- [11] *Miscellaneous Image Transformations*. URL: http://docs.opencv.org/modules/imgproc/doc/miscellaneous_transformations.html#floodfill (Odwiedzone 18. 03. 2014).
- [12] *More Morphology Transformations*. URL: http://docs.opencv.org/doc/tutorials/imgproc/opening_closing_hats/opening_closing_hats.html#closing (Odwiedzone 18. 03. 2014).
- [13] *Musescore – Download Statistics: All Files*. URL: <http://sourceforge.net/projects/mscore/files/stats/timeline?dates=2004-05-13+to+2014-08-25> (Odwiedzone 25. 08. 2014).
- [14] *MuseScore – Statistics for ftp.osuosl.org (2014-08) – main*. URL: <https://awstats.osuosl.org/reports/ftp.osuosl.org-musescore/2014/08/awstats.ftp.osuosl.org-musescore.html#month> (Odwiedzone 25. 08. 2014).
- [15] *MuseScore survey 2013 results / MuseScore*. URL: <http://musescore.org/en/node/20866> (Odwiedzone 25. 08. 2014).
- [16] *Music Notation Software Review 2014 / Best Music Writing Software / Music Editing Software – TopTenREVIEWS*. URL: <http://music-notation-software-review.toptenreviews.com/> (Odwiedzone 25. 08. 2014).
- [17] *OpenCV*. URL: <http://opencv.org/> (Odwiedzone 18. 03. 2014).
- [18] J.K. Lasocki. *Podstawowe wiadomości z nauki o muzyce*. Polskie Wydawnictwo Muzyczne, 1960.
- [19] *Qt Project*. URL: <http://qt-project.org/> (Odwiedzone 18. 03. 2014).
- [20] *Sheet music sharing / MuseScore.com*. URL: <http://musescore.com/> (Odwiedzone 05. 05. 2014).
- [21] J. Habela. *Slowniczek Muzyczny*. Polskie Wydawnictwo Muzyczne, 1988.
- [22] *The 2014 NAMM Global Report*. 2014. URL: http://www.namm.org/files/ihdp-viewer/global-report-2014/A7352D4907B25A95B2CE27A075D3956F/2014MusicUSA_final.pdf.

- [23] D. Bainbridge i T. Bell. „The Challenge of Optical Music Recognition”. W: *Computers and the Humanities* 35.2 (2001), s. 95–121. doi: [10.1023/A:1002485918032](https://doi.org/10.1023/A:1002485918032). URL: <http://www.springerlink.com/content/x1nv36548113k51u>.
- [24] *The Gamera Homepage*. URL: <http://gamera.informatik.hsnr.de/> (Odwiedzone 18. 03. 2014).
- [25] *The Music Notation Project*. URL: <http://musicnotation.org/> (Odwiedzone 25. 08. 2014).

Spis rysunków

1.1	Tabela nut używanych obecnie [9]	4
1.2	Raport stowarzyszenia NAMM z roku 2014 dot. oprogramowania do edycji notacji muzycznej [22]	5
1.3	Tabulatura gitarowa i odpowiadający zapis nutowy	6
1.4	Notacja muzyczna zapisana w kodzie Braille'a [2]	6
2.1	Klucze muzyczne na przestrzeni wieków [18]	12
2.2	Klucze muzyczne na przestrzeni wieków cd. [9]	12
2.3	Tabulatura lutniowa z XVI wieku [18]	12
2.4	Tabulatura organowa z XVI wieku [18]	12
2.5	Pismo nutowe na przestrzeni dziejów. (1. Neumy z IX wieku. 2. Neumy z X–XI wieku. 3. Kwadratowe neumy rzymskie. 4. Pismo z XV wieku. 5. Pieśń z XVI wieku. 6. Pieśń XVII wieku.) [18]	13
2.6	Notacja neumatyczna (neumy) [10]	15
2.7	Notacja modalna [10]	15
2.8	Notacja menzuralna [9]	15
2.9	Notacja włoska [9]	15
2.10	MuseScore	17
2.11	LilyPond (widok z GUI Frescobaldi)	18
2.12	Finale	19
2.13	Sibelius	20
2.14	Audiveris	24

2.15 SmartScore	25
2.16 Photoscore	26
2.17 NotateMe	27
3.1 Wyniki badania dla pytań 1-4.	33
3.2 Wyniki badania dla pytania 5.	34
3.3 Architektura programu DrawNotes	37
3.4 Diagram przypadków użycia programu DrawNotes	38
4.1 Przykładowe ręcznie narysowane symbole nut	40
4.2 Diagram klas programu DrawNotes	42
4.3 Wynik eksperymentalnej klasyfikacji odręcznie narysowanych symboli testowych	44
4.4 Przepisywany utwór — Ave Maria J.S. Bacha.	45
4.5 Próba czasowa	45

Spis tabelic

2.1	Zestawienie możliwości importu danych analizowanych programów do edycji notacji muzycznej.	20
2.2	Zestawienie możliwości eksportu danych analizowanych programów do edycji notacji muzycznej.	21
2.3	Zestawienie możliwości importu danych analizowanych programów do optycznego rozpoznawania muzyki.	28
2.4	Zestawienie możliwości eksportu danych analizowanych programów do optycznego rozpoznawania muzyki.	28

Przewodnik użytkownika

A.1 Instalacja

A.1.1 MuseScore

Należy zainstalować MuseScore z repozytorium GitHub: <https://github.com/musescore/MuseScore>. Instrukcje komplikacji znajdują się na stronie: <http://musescore.org/en/developers-handbook>. Po uruchomieniu MuseScore należy włączyć w nim serwer OSC (ścieżka Edit/Preferences/, zaznaczyć OSC remote control) i zrestartować program.

A.1.2 Gamera

Należy postępować zgodnie z instrukcjami zawartymi na stronie <http://gamera.informatik.hsnr.de/download/index.html>.

A.1.3 DrawNotes

Ściągnąć Qt ze strony <http://qt-project.org/downloads>. Otworzyć QtCreator i wczytać projekt wskazując plik qt.pro z katalogu projektu (Plik/Otwórz plik lub projekt). Zbudować projekt (kombinacja klawiszy CTRL+B) i uruchomić (kombinacja klawiszy CTRL+R).

A.2 Obsługa

A.2.1 MuseScore

Aby wykorzystać możliwości przygotowanego programu należy włączyć tryb wprowadzania nut (klawisz N).

A.2.2 Gamera

Opublikowany klasyfikator w pliku XML można edytować, dodając nowe symbole trenujące. Procedura dodawania symboli do klasyfikatora jest opisana na stronie http://gamera.sourceforge.net/doc/html/training_tutorial.html. Przygotowałem również zestaw skryptów pomocnych w tworzeniu klasyfikatora.

- convert.sh — z plików svg tworzy obrazy PNG o DPI 1200,
- montage.sh — z zadanego katalogu wypełnionego plikami PNG tworzy jeden obraz PNG,
- montage-all.sh — wykonuje skrypt montage.sh dla zadanych katalogów,
- resize.sh — zmniejsza rozmiar pliku PNG do 10%,
- resize-all.sh — wykonuje skrypt resize.sh dla zadanych plików,
- setup_wacom_tablet.sh — ustawia krzywą nacisku tabletu firmy Wacom na taką, by zawsze był rejestrowany maksymalny nacisk — przydatne podczas rysowania symboli.

Struktura katalogów symboli trenujących przedstawia się następująco:

- montaged — połączone symbole trenujące,
- png — obrazy treningowe,
 - mouse — narysowane myszką,
 - svg — wygenerowane z svg,
 - tablet — narysowane tabletem,

- resized — przeskalowane połączone symbole trenujące,
- svg — nuty w SVG,
- templates — puste pliki xcf — wzorce,
- test — pliki z symbolami testowymi,
 - big — symbole w dużej rozdzielczości,
 - small — symbole w małej rozdzielczości.

Zmontowany i przeskalowany plik można jednym poleceniem załadować do programu gamera_gui, a następnie wystarczy zaznaczyć wszystkie posegmentowane symbole i nadać im właściwą klasę.

A.2.3 DrawNotes

Na pięciolinii prezentowanej przez program można rysować symbole. Po pewnym czasie od przerwania rysowania symbole są analizowane i następuje komunikacja przez protokół OSC z serwerem OSC wbudowanym w MuseScore uruchomionym na tym samym komputerze. Program rozpoznaje akordy jako symbole rysowane koło siebie. Niestety bieżąca wersja MuseScore nie udostępnia wprowadzania akordów przez OSC, dostępne jest tylko wprowadzanie pojedynczych nut. W zakładce File/Settings można zmienić następujące ustawienia programu:

- Additional lines — liczba dodatkowych linii powyżej i poniżej pięciolinii,
- Pen point size — rozmiar pióra do rysowania,
- Lines distance — szerokość odstępu wertykalnego pomiędzy liniami pięciolinii,
- Delay to analysis — liczba milisekund bezczynności, po której następuje analiza narysowanego symbolu,
- Message timeout — liczba milisekund, w których trakcie wyświetlane są komunikaty programu na pasku statusu,
- OSC port — numer portu do komunikacji z serwerem OSC.