Expertise
and insight
for the future

Nina Schelehoff

Optical music recognition: overview, challenges, and possibilities

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Bachelor's Thesis

19.8.2020

Author Title Number of Pages Date	Nina Schelehoff Optical music recognition: overview, challenges, and possibilities 52 pages 19 August 2020
Degree	Bachelor of Engineering
Degree Programme	Information and Communication Technology
Professional Major	Software Engineering
Instructors	Antti Piironen, Principal Lecturer

The objective of this thesis is to provide an overview of OMR (Optical Music Recognition) and address the challenges and possibilities related to it. OMR is a field of research that investigates how to recognize music notation from printed and hand-written documents and to transform them into digital format. It is closely related to computer vision, machine learning, deep learning, musicology, and music information retrieval. It does not advance any of these fields, but it uses the knowledge they provide. Thus, OMR focuses, for example, on specifying what kind of information can be retrieved from music notation, how the retrieval is to be designed and executed, and what the constraints related to specific forms of notation are.

Advances in OMR contribute, for instance, to the preservation of cultural heritage, music education, music composition and practice, as well as research in musicology. In practice, some possible applications include the creation of searchable music databases for musicological analysis, the publication of archived music scores, and especially the development of software for the automatic recognition of printed and handwritten music notation as well as the encoding of the output into formats such as MusicXML, MIDI or MEI.

OMR has been researched for decades but still no computer system is able to overcome all the challenges related to music recognition. These challenges are directly related to the complexity of music notation and the lack of effective methods and algorithms capable of dealing with these problems. Music notation has evolved over the centuries into a sophisticated visual language that has its own vocabulary, syntax, and semantics, and as any other language it also has its own practices, dialects, and styles. Furthermore, developments in music continuously introduce new forms of expression. Music notation also encompasses a vast amount of music symbols that are interpreted differently depending on the context and circumstance they are presented in as well as the relationship they have to other symbols.

Due to the lack of proper technologies to solve these challenges, OMR has mainly focused on solving very specific and well-defined problems that serve limited purposes in music recognition. Nonetheless, recent advances in machine learning and deep learning could greatly improve the recognition process in the future. All in all, end-to-end OMR systems are still to be considered a problem to be solved.

Keywords	Optical music recognition, music notation, computer vision, machine learning, deep learning

Tekijä Otsikko Sivumäärä Aika	Nina Schelehoff Optinen musiikin tunnistus: yleiskatsaus, haasteet ja mahdollisuudet 52 sivua 19.8.2020
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tieto- ja viestintätekniikka
Ammatillinen pääaine	Ohjelmistokehitys
Ohjaajat	Yliopettaja Antti Piironen

Tämän opinnäytetyön tavoitteena on luoda yleiskuva OMR:stä (Optical Music Recognition) sekä käsitellä siihen liittyviä haasteita ja mahdollisuuksia. OMR on tutkimusalue, joka tutkii kuinka tunnistaa tietojenkäsittelyn menetelmin musiikkia sekä käsinkirjoitetuista, että painetuista asiakirjoista. Tutkimusalueena se liittyy läheisesti konenäköön, koneoppimiseen, syväoppimiseen, musiikkitieteeseen ja musiikkitietojen hakuun. OMR ei sinänsä edistä näitä tutkimusalueita vaan käyttää niiden tarjoamaa tietoa ja osaamista. Näin ollen OMR esimerkiksi määrittelee, millaista tietoa voidaan hakea musiikkiasikirjoista, miten haut tulisi suunnitella ja toteuttaa, sekä etsii ratkaisuja ajankohtaisiin tunnistamiseen liittyviin ongelmiin.

OMR edesauttaa kulttuuriperintömme ylläpitämistä ja edistää esimerkiksi musiikkikasvatuksessa, säveltaiteessa ja musiikintutkimuksessa käytettävää tietotekniikkaa. Käytännön tasolla OMR luo edellytyksiä esimerkiksi laajojen musiikitietokantojen sisällön analysoinnille, tarjoaa mahdollisuuksia arkistoitujen nuottikirjoitusten julkaisulle esimerkiksi MusicXML-, MIDI- ja MEI-formaatissa, sekä erityisesti keskittyy luomaan malleja ja käytäntöjä, joiden avulla voidaan kehittää ohjelmistoja automaattisen musiikintunnistuksen tarpeisiin.

OMR:ää on tutkittu vuosikymmenien ajan, mutta vielä ei ole pystytty kehittämään järjestelmää, joka selviytyy kaikista musiikin tunnistamiseen liittyvistä haasteista. Nämä haasteet puolestaan johtuvat nuottikirjoituksen monimutkaisuudesta sekä siitä, että ei ole olemassa riittävän tehokkaita menetelmiä ja algoritmeja, joilla ratkaista nuottikirjoitukseen liittyvää kompleksisuutta. Musiikki on kehittynyt vuosisatojen saatossa monimutkaiseksi visuaaliseksi kieleksi, jolla on oma sanasto, syntaksi ja semantiikka. Kuten muissakin kielissä, myös sillä omat vivahteet, murteet ja tyylilajit. Lisäksi musiikin jatkuva kehitys tuo mukanaan uusia ilmaisumuotoja. Nuottikirjoitus sisältää myös suuren määrän yksittäisiä musiikkisymboleja, jotka asiayhteyden ja tilanteen mukaan tulkitaan eri tavalla ja joiden merkitys vaihtelee sen mukaan, mihin muuhun symboliin nämä ovat liitettynä.

OMR on vuosien ajan pääasiassa keskittynyt musiikin tunnistamiseen liittyvien tarkoin määriteltyjen ongelmien ratkaisemiseen. Kuitenkin viimeaikainen kehitys kone- ja syväoppimisessa mahdollistanee tehokkaammat ja täsmällisemmät menetelmät musiikin tunnistusta varten.

Avainsanat	Optinen musiikin tunnistus, nuottikirjoitus, konenäkö, koneoppiminen, syväoppiminen

Contents

List of Abbreviations

AI	Artificial intelligence. A field of research focused on the simulation of human intelligence in computers.
ASCII	American standard code for information interchange. A character encoding standard for electronic communication.
CPDL	Choral public domain library. Virtual music score library focused on choral and vocal music within the public domain.
CWMN	Common Wester Music Notation. System for the visual representation of music that emerged in the Middle ages and develop over the centuries in the western countries. The terms Conventional Western Music Notation, Common Music Notation and Traditional Music Notation are as well used.
DCG	Definite clause grammars. A way of describing grammars for natural and formal languages in a logic programming language such as Prolog.
DIAMM	Digital image archive of medieval music. Virtual music score library focused on medieval and early modern polyphonic music manuscripts.
DL	Digital library. Online database focused on digital media content such as text, images, audio or video.
GIF	Graphics interchange format. A bitmap image format that uses irreversible compression.
HMM	Hidden Markov model. A statistical Markov model that assumes the system to be modelled to be a Markov process. Markov models are used to model randomly changing systems.
IMSLP	International music score library project. Virtual music library focused on public domain music scores.

MEI	Music encoding initiative format. Open source initiative focused on the creation of a system for the representation of music scores in digital format.
MIDI	Musical instrument digital interface. Technical standard that describes the communication protocol for the connection of electronic musical devices.
MusicXML	Music extensible mark-up language. XML based format for the representation of music notation.
NIFF	Notation interchange file format. RIFF or RIFX based format for the representation of music notation. This format has been replaced by MusicXML.
NN	Neural network. An algorithm model loosely based on biological neural networks designed for pattern recognition.
OCR	Optical character recognition. Process for the conversion of printed of handwritten text into digital format.
OMR	Optical music recognition. Field of research that investigates how to convert printed or handwritten music notation into digital format..
PDF	Portable document format. A file format used for text formatting and images. Supports vector and raster images.
PNG	Portable network graphics. A raster image format that supports lossless data compression.
PPI	Pixels per inch. A measurement of pixel density or resolution of an image.
RLE	Run-length encoding. A simple form of lossless data compression in which sequences of repeating input data is typically encoded into two bytes.
SMuFL	Standard music font layout. Standard for the mapping of musical symbols optimised for the modern font formats.

TIFF	Tagged Image File Format. Image file format which can be compressed or uncompressed and is used to save raster images. Widely supported for scanning, image manipulation, word processing, OCR, and OMR.

1 Introduction

Music is a universal language that already existed before spoken language. It seeks to communicate information and convey emotions. Music happens in real time. The minute it stops, it belongs to the past unless there is a way to preserve it. For thousands of years, music was entirely an oral tradition. However, approximately a thousand years ago, music started evolving into a literate tradition, and nowadays, there are numerous ways of recording music with electronic devices. Nonetheless, before these devices were invented, humans preserved music via handwritten notation in order to reproduce it later. As such, music notation is linked to the human ability to read and write, and millions of songs have been documented over the past centuries.

Interest in the conversion of music scores into digital format has been steadily growing over the past decades. The importance of creating computer-based systems for the recognition of music notation lies in the preservation of people's cultural heritage and the facilitation of musicians as they still today often choose to record their work with traditional pen and paper. It is also worth observing that a large part of historical music manuscripts still exists only in unpublished paper format. Thus, finding an effective way for automating the digitation of music scores would save a lot of human effort.

The objective of the thesis is to give an overview of optical music recognition (OMR) and assess its possibilities and challenges. OMR is a field of research that investigates how to convert printed of handwritten music notation into digital format. From the technical perspective, it is considered a subfield of computer vision and document analysis, and it is also closely connected to machine learning and especially deep learning. From the musical point of view, it is related to digital musicology, music composing and practice as well as music information retrieval. Advances in the field of OMR can be applied to areas such as the creation of large-scale searchable databases for musicological analysis, the editing of existing notation, music teaching, publication of archived music scores, or the further conversion of music scores into file formats such as MIDI, MusicXML or MEI. From the practical point of view, the benefit of OMR lies in the reduction of costs related to digitation.

The challenges related to OMR are related to the complexity of music notation itself. In principle, the notation is complex because of the vast amount of information it tries to

convey and the variations in notation styles that have evolved throughout history. This thesis focuses specifically on the Western tradition of music notation which had its first breakthrough in the Middle Ages. Over the centuries, it has evolved into a sophisticated visual language that has its own syntax and semantics. The full understanding and further development of OMR requires, in addition to technical knowledge, a good understanding of the principles and intricacies of music notation as well as how it became to be what it is today and how it was interpreted in the past. Music history and music theory as such go beyond the scope of this thesis, but some of the main aspects that need to be taken into consideration in OMR are discussed in the first chapters of this thesis.

2 Western music notation

2.1 Breakthroughs in music notation – historical background

Western music notation started evolving in the Middle Ages. It continued developing in the Renaissance, and it basically achieved the form that is well known today in the Baroque period. What started as simple instructions on the general melodic shape of a song grew into something as complex as the detailed instructions on how to perform a piece of music for an entire symphony orchestra. (Kelly, 2015.)

The earliest documented music notation was liturgical, vocal, and monophonic which means that it was performed by one singer or a chorus in unison without accompaniment. Nonetheless, pictorial, and literary references have revealed that also secular, polyphonic, and instrumental music existed at the time, but it is not a well-documented tradition. (Taruskin, 2006.) Gregorian chant from the ninth century is the first known repertoire of liturgical songs that involved a rudimentary way of music notation which consisted of signs called neumes above the text that specified the rise or fall of a melody in relation to the text (Paxman, 2014). In the late ninth century, the relative height of neumes indicated the melodic shape in more detail, but neumes still did not specify the exact pitch of notes nor their duration (Kelly, 2015). By the end of the tenth century, some scores also included one or two staff-lines to help reading the relative height of neumes (Paxman, 2014).

In the eleventh century, an Italian monk called Guido of Arezzo, invented a system of writing music that allowed the performer to sing something never heard before (Taruskin, 2006). Consequently, music started shifting from being an exclusively oral tradition to also a literate one. Guido organized notes into groups called hexachords along the lines and spaces of a color-coded four-lined stave set at intervals of a third (Paxman, 2014). He also created solmisation and gave the sequence of notes specific names which were ut – re – mi – fa – so – la. The idea behind the coloured staff-lines was that the red one indicated fa and the yellow one ut. Thereafter, it was possible to preserve pitches in relation to each other and sing a melody without knowing it beforehand (Kelly, 2015). Guido's system was precise in the vertical axis of notation, but the arrangement of notes on the horizontal axis to indicate rhythm and temporal sequence still remained imprecise.

The means to notate rhythm emerged in the twelfth century along major developments in polyphony, which was seen as a way to embellish liturgical music with several concurrent melodies. A large repertoire of polyphonic music was created especially at the Cathedral of Notre-Dame in Paris by two French composers, Leonius and Perotinius. As polyphony grew in complexity, music needed to be coordinated in a more timely manner. Leonius had already incorporated rhythmical aspects into his work, but the earliest documented music scores with explicit indication of rhythm were created by Perotinius. These pieces of music were called Viderunt and Sederunt, and rhythm was notated by grouping notes into formations called ligatures. Music created by Leonius and Perotinus was essentially a polyphonic version of Gregorian chant. (Kelly, 2015.)

The way music was notated also changed during the twelfth century. A fifth staff-line as well as a shorter note called breve were introduced, and notes were written as square-shaped symbols that occasionally had a stroke. This new way of notation was called squared notation, and it emerged as a consequence of writing with a new kind of pen that had a broad end. Depending on the position of the pen, the stroke was either thick or thin. (Kelly, 2015.)

In the thirteenth century, music notation saw more innovations especially in terms of rhythm. Franco de Cologne meticulously described these new breakthroughs and eventually the newly emerged style of notation was referred to as the Franconian style (Kelly, 2015). The most significant developments included the indication of the length of rests in music scores, and the indication of the length of each individual note with a specific shape, thus making ligatures obsolete (Taruskin, 2006). Franco also invented a diamond shaped note called semibreve that was shorter than the breve, and a system for the indication of time measure. (Paxman, 2014.) It is worth mentioning that even though the Franconian style became extremely popular, there still were musicians who preferred the old way of notating music. For example, the French troubadours and the French trouvères still mostly notated music in the Gregorian chant style (Kelly, 2015).

In the fourteenth century, Europe became devasted by endless wars and catastrophic plagues. Despite all adversities, as illustrated in Figure 1, music notation continued evolving and ultimately Philippe de Vitry developed a system that forms the foundation of western contemporary music notation (Taruskin, 2006). De Vitry's key innovations include a new way of systematically dividing notes, the creation of a new smaller note than

the semibreve called the minim, the invention of the double time measure, and the colour-coded indication of changes in rhythm (Kelly, 2015).

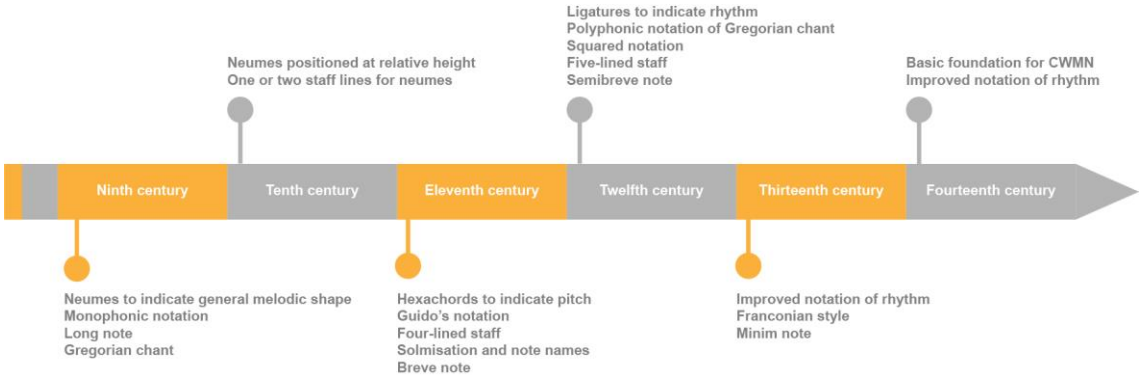

Figure 1. Summary of major breakthroughs in music notation.

Advances in music notation until the fourteenth century made it possible to record pitch and rhythm so that songs could be sung and played without knowing them previously. Hence, the oral tradition of preserving music shifted to a literate tradition. Music notation still continued evolving especially during the renaissance and the baroque eras, but innovations were more related to expression and articulation, and from this perspective were less dramatic than in the Middle Ages (Kelly, 2015).

2.2 The current complexity of CWMN

Over the centuries, CWMN has evolved into a sophisticated visual language that has its own vocabulary, grammar and syntax, and as any other language, it also has its common practices, dialects, and styles that make it semantically complex (Feist, 2017). It is worth observing that SMuFL which currently forms the foundation for music font mapping, includes over 2,400 music symbols and hundreds of optional symbols for different historical periods, types of music and instruments (SMuFL, 2019). Furthermore, the development of existing music symbols still continues along with the introduction of new forms of musical expression. This large number of music symbols found in printed scores together with their variance in size and shape in handwritten scores, and especially in historical manuscripts makes OMR extremely challenging (Rebelo et al., 2012).

In addition to the complexities related to the different ways of notating music throughout history and the vast amount of music symbols, challenges are also related to individual music symbols as well as how symbols relate to each other (Bainbridge & Bell, 2001).

2.2.1 Individual music symbols

In essence, CWMN expresses pitch, time, loudness and timbre which are indicated by a series of notes and additional symbols arranged on five horizontal staff-lines or the spaces in between which are called staff-spaces (Feist, 2017). The stave refers to the group of staff-lines, and it is divided into measures or bars which in turn are delimited by barlines (Gould, 2014). Pitch is represented by notes on the vertical axis of the stave. Nowadays there are seven notes and they are named after the alphabet: A, B, C, D, E, F, G. Notes that are an octave apart from each other sound similar and thus have the same name, but they are written either with capital or small letters or with a superscript index (Joutsenvirta & Perkiömäki, 2014). As illustrated in Figure 2, the stave is often extended to pitches outside the range of staff-lines by using ledger lines (Feist, 2017).

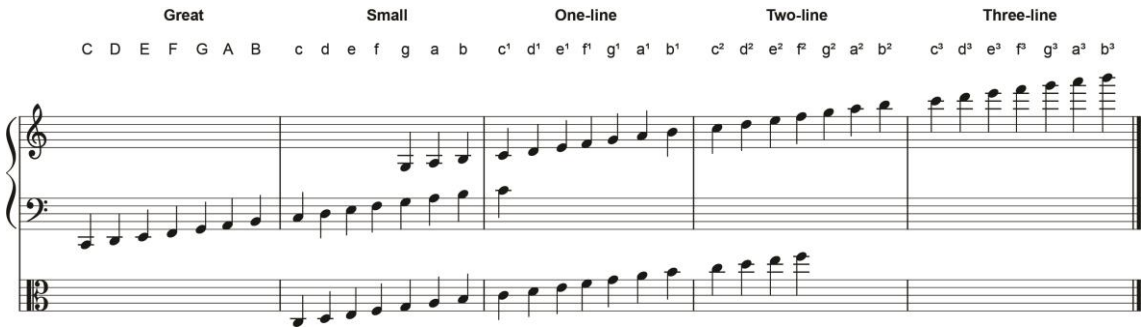

Figure 2. Notes and their names displayed on staves and ledger lines according to the designated clef.

The clef indicates the relative position of a certain pitch on the stave. The most common clefs in contemporary music are the G or treble clef, the F or bass clef, the C or alto clef, and the tenor clef. Over the past centuries, a great variety of other clefs have also been used. Nowadays, from the range of C clefs only the viola and the trombone use the alto clef, and the tenor clef is used by the bassoon, trombone, cello and occasionally the double bass (Gould, 2014). The rest of the C clefs were common until the eighteenth century in choral music (Joutsenvirta & Perkiömäki, 2014).

A note on the same staff-line or staff-space can have different pitches depending on the clef preceding it. There are eight types of G-clefs which are treble, treble 8va alta, treble 15 ma alta, treble 8va bassa, treble 15 ma bassa, double treble 8va bassa, treble 8va bassa, and the French violin. Figure 3 depicts the position of the g^1 note on the vertical axis.

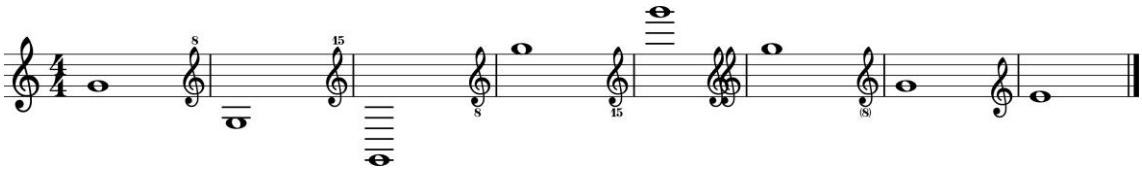

Figure 3. G-clefs that depict the position the g^1 note on the stave.

There are seven types of F-clefs which are bass, bass 8va alta, bass 15ma alt bass 8va bassa, bass 15ma bassa, baritone, and subbass. Figure 4 shows the location of the f note on the vertical axis of the stave.

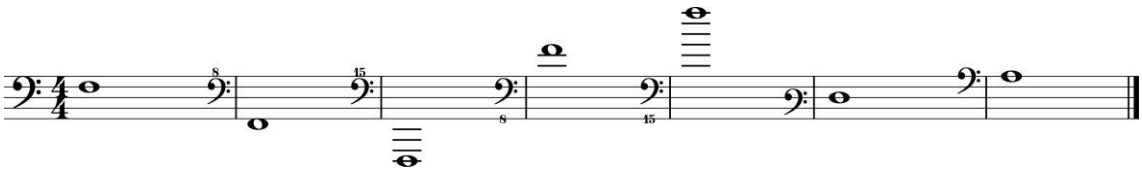

Figure 4. F-clefs that depict the position of the f note on the stave.

The most common C-clefs include the soprano, mezzo-soprano, alto, tenor and baritone clefs. As show in Figure 5, they all indicate the position of the c^1 note on the vertical axis of the stave. The French violin clef is very rare today. It was mostly used in France in the baroque era. This variety of pitch location on the stave has to be taken into account during the optical recognition of music scores.

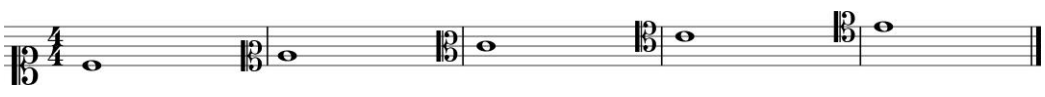

Figure 5. C-clefs that depict the position of the c^1 note on the stave.

Accidentals or chromatic symbols indicate that a note has to be raised or lowered by a semitone or a tone. The most common accidentals are the sharp and the double sharp which raise a note respectively by either a semitone or a tone, the flat and double flat

which respectively lower a note by a semitone or a tone, and the natural sign which indicates that a raised or lowered note is to be restored to its original pitch (Feist, 2017). It is also worth noting that each accidental affects only notes of the same diatonic pitch, in the same octave and clef, and until the end of the measure, it has been placed on (Gould, 2014).

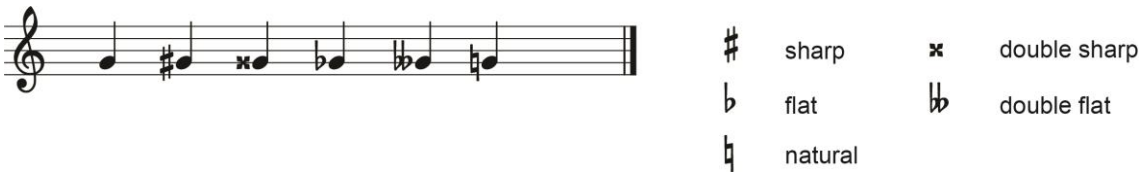

Figure 6. Common chromatic symbols or accidentals used in CWMN.

From the perspective of individual music symbols, the use of accidentals poses two challenges to OMR. Firstly, their use has been inconsistent over the centuries and often a matter of the composer's or writer's taste. The natural symbol was first met in the twelfth century and the sharp symbol was introduced in the thirteenth century. However, the meaning we assign today to the natural symbol was not encountered until the eighteenth century (Joutsenvirta & Perkiömäki, 2014). Secondly, the same pitch can be graphically represented on the stave either by using the flat or the sharp symbol. For example c^1 sharp has in reality the same pitch as d^1 flat even though they can be depicted in alternative ways as indicated in Figure 7 (Joutsenvirta & Perkiömäki, 2014).

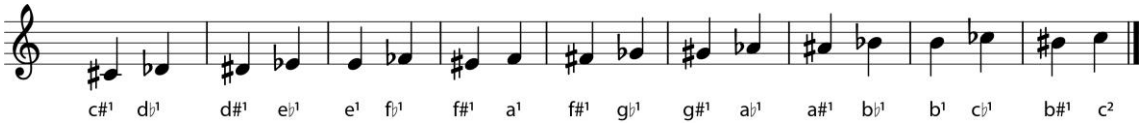

Figure 7. Examples of different forms of notation that indicate the same pitch.

Key signatures are groups of sharps and flats positioned at the beginning of the stave after the clef and before the time measure to indicate notes to be raised or lowered until the end of a music score or until another key signature is encountered. In contrast to accidentals, key signatures affect pitches in all octaves. Their main objective is to reduce the amount of needed ledger lines and thus decrease complexity within the music score (Gould, 2019). Key signatures form diatonic scales with a specific pattern that indicate the modality (major or minor key) of a piece of music (Feist, 2017). The recognition of the tonic or first note and the subdominant or fifth note of the scale contributes to the

recognition of altered notes which are notes outside the scale used to provide a certain tension to a piece of music (Joutsenvirta & Perkiömäki, 2014).

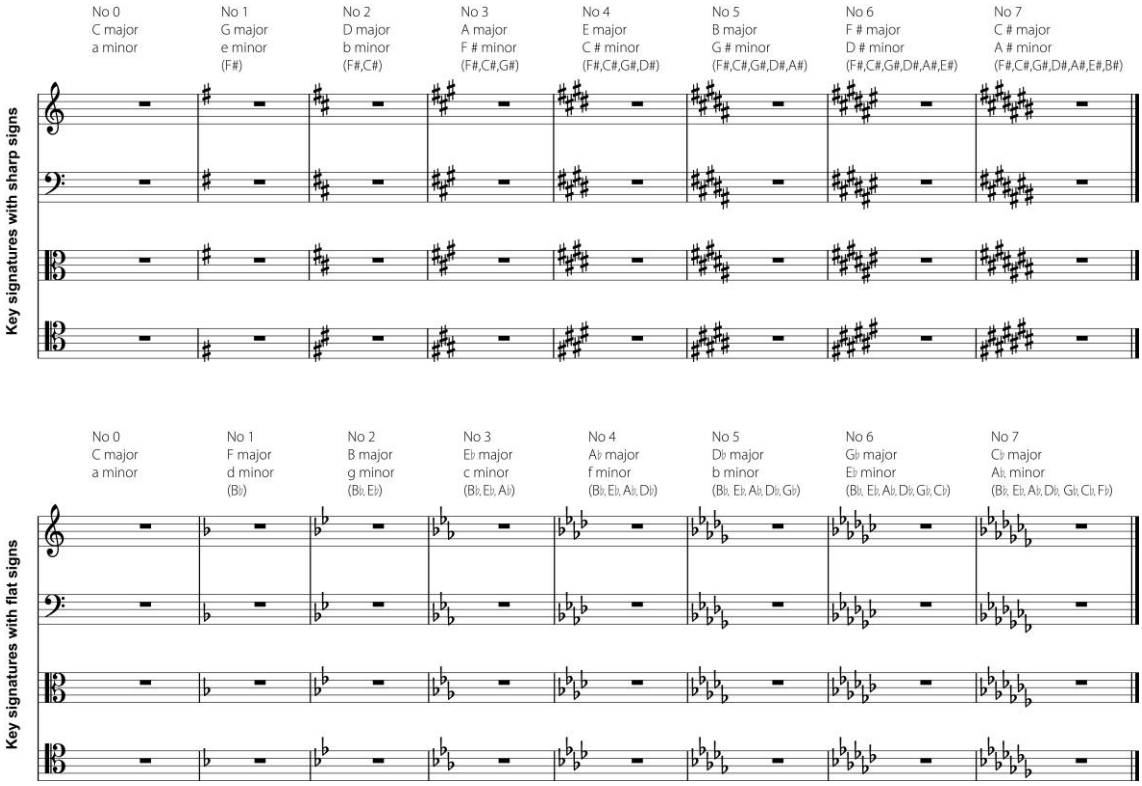

Figure 8. Key signatures with sharp and flat signs.

As shown in Figure 9, chords are usually formed by combining two or more notes of the same duration onto the left side of the same stem (Feist, 2019). It is worth observing that in string music notes with different duration can be attached to the same stem (Gould, 2014). Usually the direction of the chord's stem is determined by the furthest note from the centre of the stave; the stem goes downward when the furthest note is above the centre; and downward when it is below (Feist, 2019). Chords pose a challenge to OMR because noteheads can be positioned on any point along the stem and they can even be pushed to the right side of the stem to avoid overlapping for example when writing intervals of a second, or when the chord has an odd number of notes (Pacha, 2019). Thus, the amount of graphical variations for chords is theoretically so high that a set of pre-set templates cannot be used for matching graphical shapes and chords (Bainbridge & Bell, 2001).

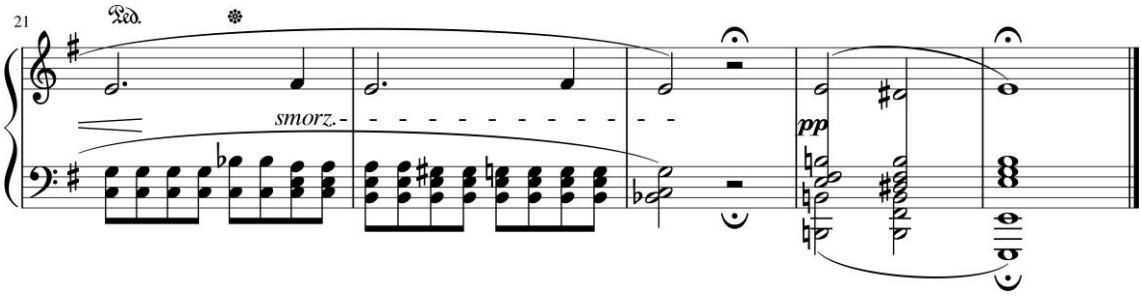

Figure 9. Noteheads placed along the same stem (Chopin, 1829-32).

Another challenge related to individual music symbols is that the same symbols can be depicted in more than one way due to musical requirements (Bainbridge & Bell, 2001). This happens, for example, with flagged notes that have been grouped together by beams to increase readability (Byrd, 1984). These variations include, for example, the vertical stretching of stems to connect them to the beam (normally the length of the stem is one octave), the horizontal stretching of beams due to the vertical alignment of other voices played at the same time, the shearing of beamed notes, and the rotation of notes due to their placement on higher or lower staff-lines (Bainbridge & Bell, 2001). Figure 10 shows an example which depicts the modification of beamed notes by horizontal and vertical stretching as well as shearing. Notes positioned on the third staff-line and above it, usually have downstems, while notes placed below the third staff-line have upstems. Accordingly, downstem notes have flags that curve outward and upstem notes have flags that curve inward (Feist, 2017). It is worth observing that in the case of sheared beamed notes, the angle is often determined by the music setter and it often creates a distinctive visual characteristic to the score (Gould, 2014).

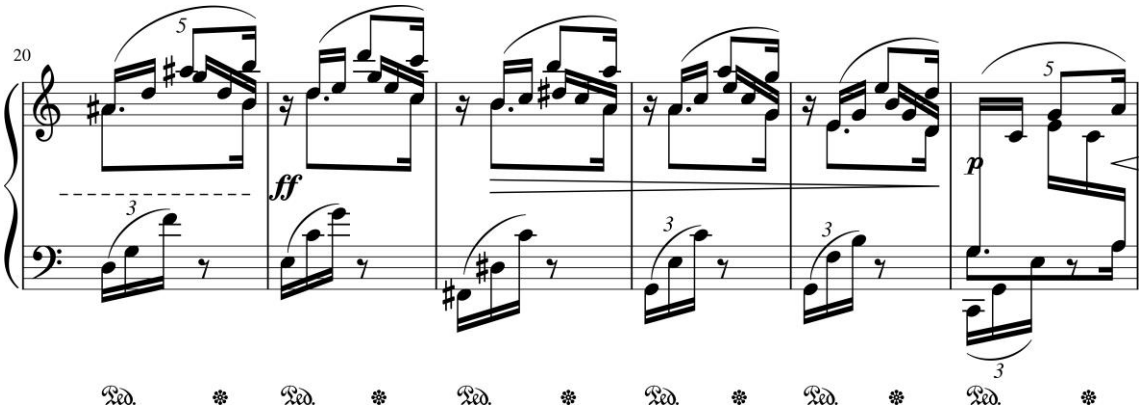

Figure 10. Horizontal and vertical stretching and shearing (Chopin, 1838-19)

Minor additions to music symbols may reflect important information in terms of musical expression. These additions include for example the use of articulation marks to clarify how a single note ought to be played (Bainbridge & Bell, 2001). Articulation marks are particularly ambiguous for OMR as they ought to be interpreted according to the style and period of the music score in question (Gould, 2014). Nowadays, however, the general interpretation is that notes with a staccato mark should be played shortly, notes with an accented mark should be played loudly, notes with a tenuto mark should be played long, and notes with a strong accent mark should be played very loudly (Feist, 2017). Articulation marks are placed either above or below the notehead depending on its direction. During the recognition process, staccato marks should not be confused with dotted notes which normally have one or two dots on the right-hand side of the notehead to express an extension in the duration of the note (Gould, 2014).

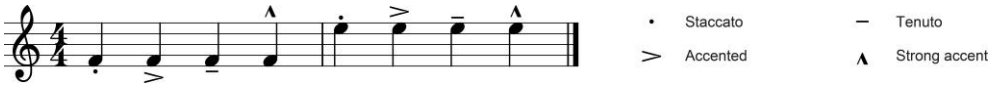

Figure 11. Common articulation marks placed either above or below the note head.

The vertical position of the dot or dots depends on the whether the notehead is on the staff-line or on the staff-space; when the notehead is on the staff-space, the dot is aligned with the middle of the notehead; and when the notehead is on the staff-line, the dot is in the middle of the subsequent staff-space (Byrd, 1984). Notes placed along the horizontal axis of the stave follow time intervals but the exact distance between notes cannot be explicitly specified because other music symbols such as dots and accidentals often push notes further than otherwise necessary (Gould, 2014).

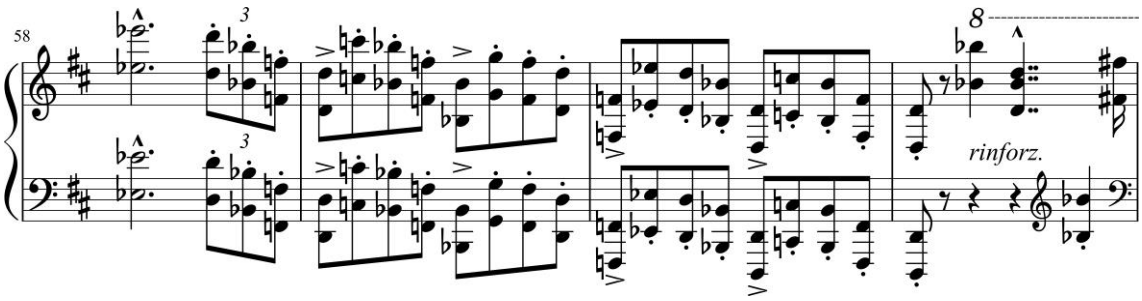

Figure 12. Dotted notes and articulation marks above noteheads (Schumann, 1841-42).

2.2.2 The relationship between individual symbols

OMR can be seen as an extension of OCR which is an area of research that focuses on the recognition of printed and handwritten text from documents (Bainbridge & Bell, 2001). OCR emerged in the 1940s and it was developed to recognise text from large amounts of paper such as government records, credit card imprints, commercial forms, and addresses on envelopes (Fornés, 2005). As such OCR is not applicable to music notation because the graphical properties of music symbols are very different from the properties of letters. Whereas text is organized along the page conforming to a baseline on the horizontal axis, the vertical axis is used in a very simple way. In music scores on the other hand, the use of the vertical axis has been extended to depict pitch (Bainbridge & Bell, 2001). Hence, OCR is applied to music notation only for the recognition of text such as lyrics and dynamic markings (Fornés, 2005). Nevertheless, the conversion of letters into ASCII form is not sufficient because text on a score is always associated to certain parts of the composition and therefore its meaning depends on its precise location (Bainbridge & Bell, 2001).

The difference between text and music notation is not limited to the variety and alteration of its individual symbols (Bainbridge & Bell, 2001). Whereas text is one-dimensional in layout, music is two-dimensional which means that the interpretation of music symbols is defined by their relationship to other symbols on the score (Pacha, 2019). For example, a clef at the beginning of the stave affects subsequent notes until the end of the score or until another clef is encountered; similarly, an accidental inside a bar affects subsequent notes on the same staff-line or staff-space within the same octave until the end of the bar or until another accidental is encountered (Bainbridge & Bell, 2001).

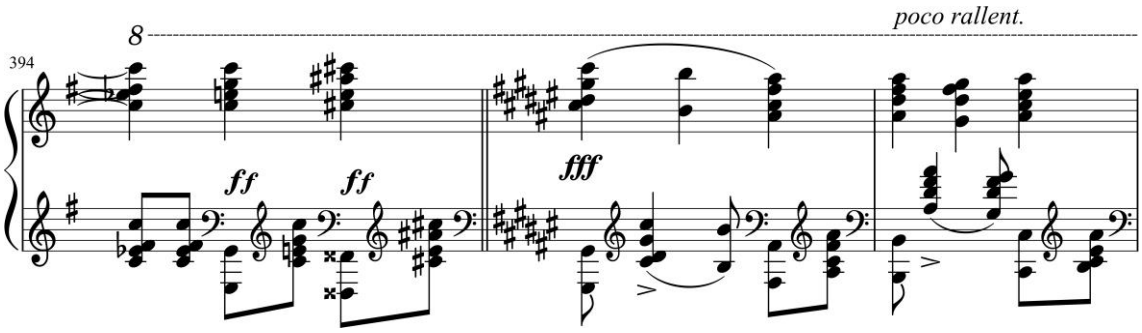

Figure 13. Change in key signature and clef (Schumann, 1841-42).

For example, as illustrated in Figure 14, the G clef at the beginning of the stave fixes the position of the g^1 note on the second staff-line and subsequently the c^2 on the third staff-space. However, the pitch of the c^2 note changes according to its relationship to other music symbols. In the first example (a), the first c^2 sharp is an altered note which affects the pitch of all notes on the same staff-space within the same octave until the end of the measure; therefore, the eighth note on the stave is also a c^2 sharp. In the second example (b), however, there is an octave sign above the stave which indicates that notes under the dashed line are to be played one octave higher; therefore, the eighth note is a c^3. In the third example (c), due to a prior change in clef, the eighth note is in a different octave and also has a different pitch, in this case it is an e. In the fourth example (d), the passage is played as at the time the song was composed which means that the eighth note is a c^2 in order to avoid an augmented interval between the note in question and the following note. (Byrd & Simonsen, 2015.)

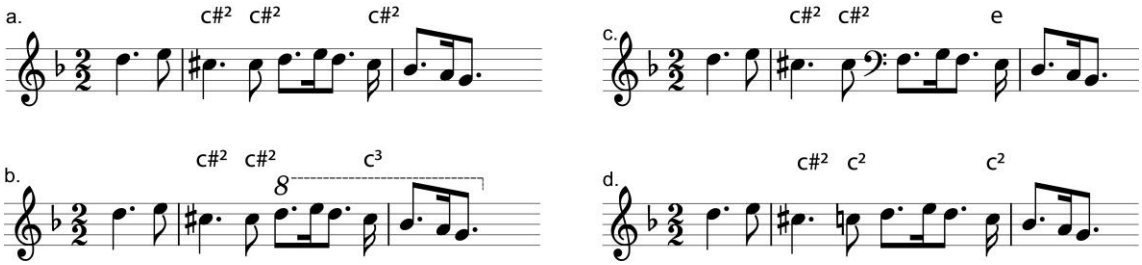

Figure 14. Changes in pitch related to the relationship to other music symbols (Purcell, 1702).

The two-dimensionality of music notation also affects the order in which notes are played (Bainbridge & Bell, 2001). In single voice music, this usually does not pose a problem, but for example piano music is written on two staves, one for the right and one for the left hand (Gould, 2014). Thus, sometimes the exact hand distribution may seem ambiguous on the score and experience in piano music is needed to intuitively recognise the correct order in which to the play the notes (Pacha, 2019). However, scores for multiple instruments, such as the string quartet as depicted in Figure 15, are played simultaneously (Bainbridge & Bell, 2001).

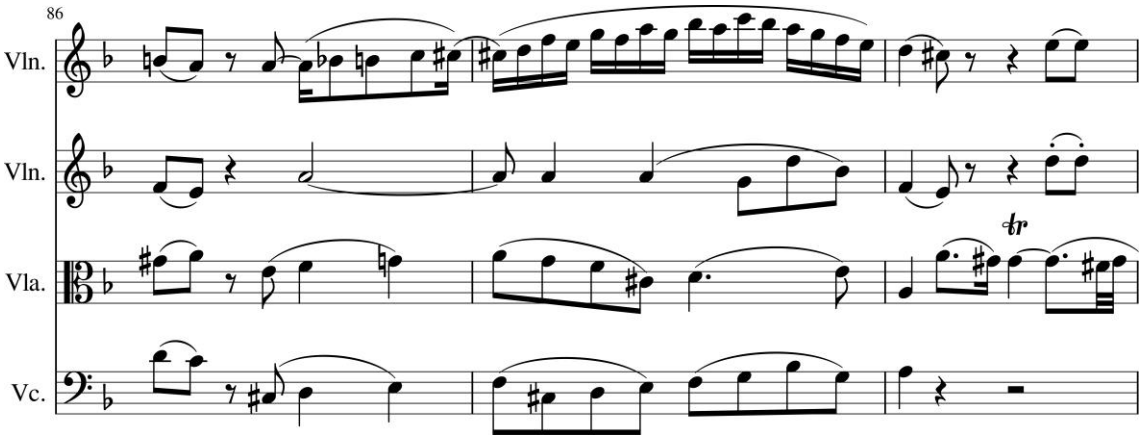

Figure 15. Music score for a string quartet (Mozart, 1783).

Even in a simple situation when the correct order for playing music is evident, the computer will struggle in the determination of the rules related to order. For example, as depicted in Figure 16, the computer has to determine which symbols to process first, the beamed notes or the accidental. According to music theory, the second note and the accidental form a subgroup, but the computer would connect the beamed notes as a group because they are physically connected, and the chromatic symbol would be recognised next and in isolation (Bainbridge & Bell, 2001).

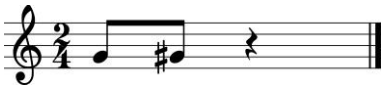

Figure 16. The effect an accidental has on a beamed note.

Some music symbols pose exceptional challenges to OMR and such is the case with slurs as they are represented by long randomly shaped curved lines that connect notes (Novotný & Pokorný, 2015). The problem is related to their arbitrary shape which means that their location, longitude, and height can basically be anything. Slurs have different meanings for different instruments, but usually they indicate notes that should be played without separation (Feist, 2017). For example, in string music they indicate notes to be played with one bow stroke, and in vocal music notes to be sung in one syllable (Gould, 2014). During the recognition process, slurs should not be confused with ties which are similar graphical symbols, but indicate the absence of rearticulation (Feist, 2017). Figure 17 shows an example of a stave with slurs and ties.

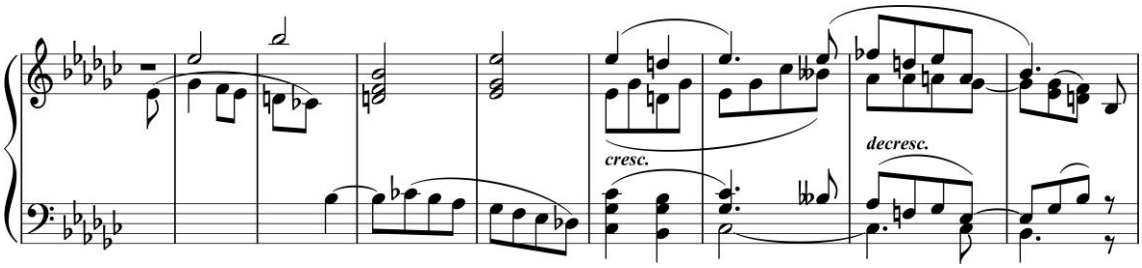

Figure 17. Slurs and ties on the same music score (Chopin, 1834-35).

Computer algorithms also struggle with the recognition of superimposed symbols such as overlapping beams, or dynamic markings that cross over bar lines, or in the case of piano music, markings that intersect with staves from one hand to the other (Bainbridge & Bell, 2001). Figure 18 shows an example of overlapping stems and beams.

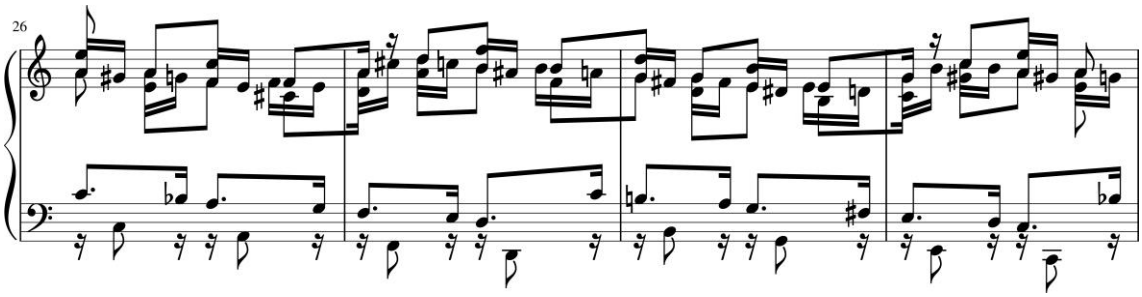

Figure 18. Overlapping stems and beams (Schumann, 1829-32).

Crescendo and diminuendo markings or hairpins are generally placed horizontally on the score, but sometimes they might be positioned as tilted markings that follow a progression of pitches (Gould, 2014). Figure 19 depicts an example of a crescendo hairpin that cuts through four barlines.

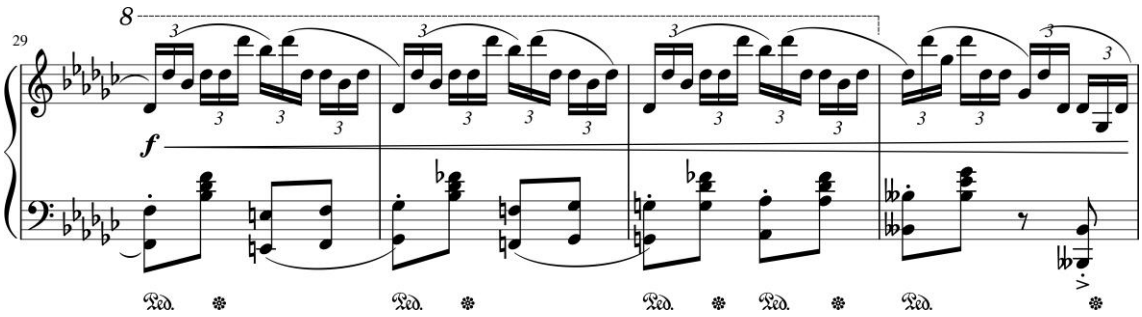

Figure 19. A crescendo marking crossing over several barlines (Chopin, 1829-32).

Markings that cross barlines and intersect several staves are also common in piano music. Figure 20 shows an example of long slurs that cross over one key to the other across staves and barlines.

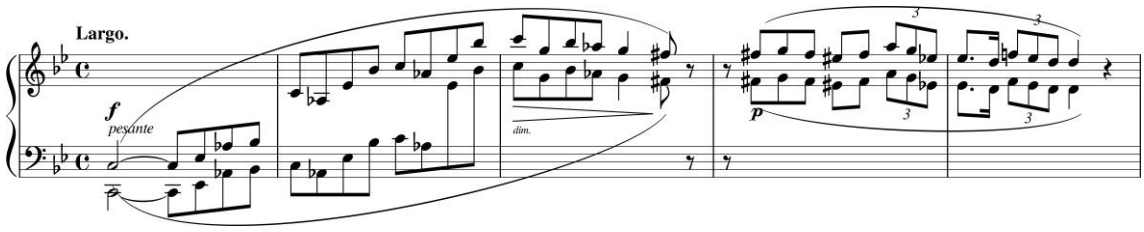

Figure 20. Slurs crossing over staves and barlines (Chopin, 1834-35).

To summarise, the most challenging aspects of OMR are related to how music notation has changed and evolved over the centuries, as well as to the complexity of music notation in terms of syntax and semantics, the adaptability of individual music symbols to specific situations, and the changing properties of music symbols according to their relationship to other symbols. Additionally, it must be considered that the rules of music are at times ambiguous and occasionally intentionally broken by the composer. From this perspective, OMR requires more than the mere identification of music notes and the recognition of their relationships optimised to a certain musical period, style of music, or instrument. However, currently, there is no computer system capable of interpreting music in such a demanding way. Once information has been disregarded or a composer breaks any of the conventions related to notation syntax or semantics in a music score OMR will produce erroneous interpretations of the notation. This is a problem that could possibly be solved to some extent with machine learning and deep learning approaches in the far future.

3 Optical music recognition defined

3.1 Relation to other fields of research

As seen in previous chapters, OMR is closely related to fields such as musicology, music composition and practice and music information retrieval. The objective of this chapter is to put OMR in context in relation to fields of research within computer science. OMR is a subfield of computer vision and document analysis, and it is also closely linked to research in artificial intelligence, machine learning and deep learning (Calvo-Zaragoza et al., 2019).

In its simplest form, computer vision refers to the field of research that concentrates on the automated extraction of graphical information from electronic documents. The retrieval of information can include tasks such as object detection, recognition, and grouping, as well as image content search and examination (Solem, 2012). Computer vision has been extensively researched; nonetheless, the computer still cannot interpret an image as a human would do. Szeliski (2010, page 3) aptly describes the underlying difficulty as follows: “vision is an inverse problem, in which we seek to recover some unknowns given insufficient information to fully specify the solution”. Thus, modelling our visual realm and all its complexities is a difficult endeavour, and this can be said to apply to OMR as well. From a broader perspective, computer vision refers to the field of research that aims at describing the content of an image and the reconstruction of its particular features (Szeliski, 2010). Despite its intrinsic difficulties, computer vision is still largely studied and other fields of research such as medical imaging, automotive safety, surveillance, biometrics, and OCR use its outputs. OMR also widely uses computer vision methods and algorithms for the detection and identification of music objects on the score. Moreover, along the advances in machine learning and deep learning, new emerging approaches will highly likely prove to be valuable in the field of OMR, especially when appropriately combined with methods and algorithms of computer vision.

AI is a general field of computer science that includes machine learning and deep learning as subfields. AI also encompasses symbolic AI which does not include any learning as such, but hardcoded rules imposed by programmers. Symbolic AI is a suitable approach for solving well-defined problems, but it is unsuitable for multifaceted problems

that involve tasks such as image classification and interpretation. The purpose of machine learning is to go beyond and make computers learn on their own to solve complex problems. The basic idea is that the computer learns the rules associated with a problem instead of having programmers hardcode the rules. Hence, systems based on machine learning must be trained to solve problems. The training is carried out by providing the computer a meaningful representation of the input data and examples of expected outputs. A way is also needed to measure that algorithms eventually perform as intended. Deep learning takes machine learning even further by emphasizing the learning of successive layers of meaningful representations which are learned by using models called neural networks. The term deep refers to these successive layers of meaningful representations. Image manipulation, in turn, typically uses convolutional neural networks which consist of an input layer and an output layer as well as several hidden layers called convolutional layers. (Chollet, 2018.)

Deep learning has been successful in areas such as image classification, speech recognition, text recognition, language translation, and handwriting recognition (Chollet, 2018). Even though complete end-to-end systems have successfully been developed by using deep learning in other fields of research, currently there are no complete end-to-end OMR systems that can transform a music score into encoded music as an output (Pacha, 2019). However, advances in the creation of end-to-end systems in deep learning and OMR have been made. As a start, convolutional neural networks have been successfully trained to identify music notation from other images within a database of 2,000 images and with an accuracy of nearly 100% (Pacha, 2019). This automated task of identification is useful, for instance, in scenarios where the user needs to find music scores from a large set of random documents. Convolutional neural networks have also been able to successfully classify the hand-written music symbols contained in the HOMUS dataset which has 15,200 samples of 32 different hand-written music symbols written by 100 different musicians with an accuracy of 96% (HOMUS, 2020). However, it can be reasoned that the model still needs research as currently SMuFL contains over 2,500 different music symbols. Therefore, even though advances have been made, research is still needed to train models that can accurately recognize all music objects available in SMuFL. Research is also needed for the correct interpretation of the relationships between music symbols and on the proper assignment of musical meaning to the score without having programmers formulate explicit rules (Pacha, 2019). As seen in previous chapters, this is not a trivial task due to the complexities related to music notation. Since

one of the main objectives of OMR is the automation of tasks that require human effort such as the creation and maintenance of large searchable databases for musicological analysis or the further conversion of notation into digital format, robust and efficient OMR systems are evidently needed.

3.2 Definition of OMR

According to Calvo-Zaragoza et al. (2019), OMR has not been defined precisely. In their research, they went through a large amount of papers on OMR, encountered several definitions but found them either relatively ambiguous or restricted. Therefore, they suggest the following definition for OMR: “OMR is the field of research that investigates how to computationally read music notation into documents.” They argue that OMR is to be referred to in a broader sense as a field of research and not merely as a process, system, or technique. OMR should also be clearly differentiated from other fields of research, such as musicology, computer vision or machine learning, because OMR uses the knowledge these fields provide but it does not concentrate on advancing them. OMR also specifies what kind of information is to be recognized from music scores and how the recognition process should be designed and carried out.

Music notation systems are designed to preserve the most important information about music, but , in addition, a certain degree of either intentional or unintentional information always occurs loss while notating it (Kelly, 2015). For example, tempo or dynamic markings could have been disregarded, and even rhythm or pitch could be missing from older manuscripts. In the field of music, lost information has been left to the musician’s better knowledge or taste. Calvo-Zaragoza et al. (2019) suggest that this is also where OMR boundaries are encountered because at present the computer cannot complete pieces of missing information that require prior knowledge of music or the ability to intuitively interpret it. Nevertheless, future advances in deep learning could change the situation.

3.3 OMR inputs

The OMR input refers to the original music notation document to be recognised. It has an enormous impact on the quality of the encoded OMR output after the recognition process, and therefore the requirements and design of the algorithms that perform the

recognition. OMR inputs have been categorised in five areas which are (1) offline OMR, (2) online OMR, (3) music notation systems, (4) music notation complexity, and (5) input quality. (Calvo-Zaragoza et al., 2019.)

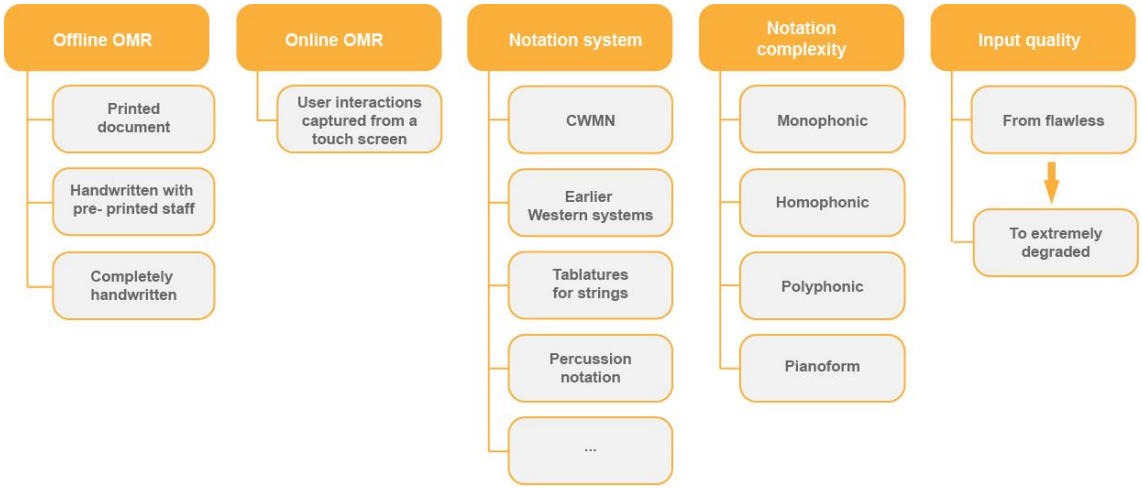

Figure 21. OMR input categories as defined by Calvo-Zaragoza et al., (2019).

The first category of OMR inputs refers to offline OMR which includes static images of music scores that can be printed, handwritten on pre-printed staff paper, or completely handwritten (Calvo-Zaragoza et al., 2019). Research on the recognition of simple printed music scores has been relatively successful compared to the recognition of handwritten music scores which introduces additional difficulties to OMR because writers have different writing styles, hand-drawn staves may appear distorted, and the spaces between staff-lines may be uneven (Fornés, 2005). Additionally, the relative size of music symbols may vary greatly as they may appear overlapping each other, or they may be positioned at varying distances from each other (Novotný & Pokorný, 2015). Historical music manuscripts pose an even bigger challenge because of paper degradation, evolving music notation, and the lack of notation standards (Fornés, 2009).

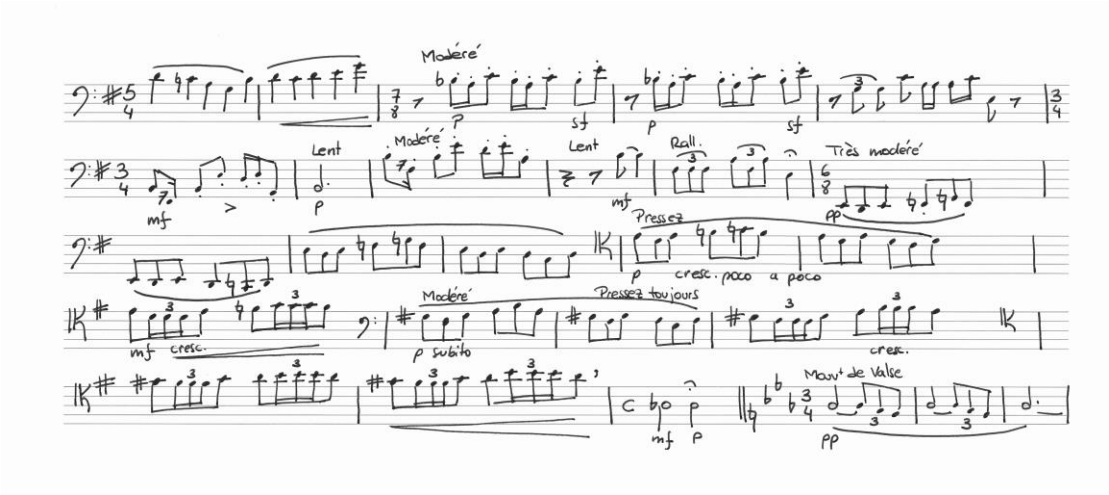

Figure 22. Sample from the CVC-MUSCIMA dataset depicting handwritten music notation written on pre-printed staff paper (Fornés et al., 2012).

Offline OMR inputs are converted into digital form before starting the recognition process. Kinser (2019) states that in its simplest form, a digital image is saved so that each pixel is stored in three bytes, one for each colour channel. This method, however, produces extremely large image files which would make the recognition process inefficient. Hence, images have to be converted to compressed file formats such as TIFF, GIF, PNG, JPEG or PDF which are all different formats of raster images. A raster image refers to a dot matrix data structure formed as a rectangular pixel grid that is used, for example, for photographs and scanned images. It is worth mentioning that a PDF file can also have a vector file format which is based on mathematical formulas that define the image.

The TIFF format can be saved with or without compression, and it is widely used for scanning, image manipulation, word processing, OCR, and OMR. The JPEG format is mainly used in photography and its compression efficiency is achieved by decreasing the clarity of the image's sharp edges. The degree in compression can be adjusted, but once the image is saved, lost information cannot be restored. The GIF format uses a palette of 256 colours and compresses the image by replacing colours from the original image with colours that best match the palette. Thus, the GIF format is suitable for gray-scale images and for images with very few colours. Once the file is saved, lost information cannot be restored. The PNG format does not loose information after having been compressed. However, its file size is bigger than in the JPEG format. (Kinser, 2019.)

Offline inputs are also often saved as PDF files which support text as well as raster and vector images. OMR input files are usually saved at a resolution of 300 ppi and the commonly supported file formats of the most relevant OMR software range from PDF, TIFF, JPEG and PNG to GIF (Rebelo et al. 2012). The quality of the input image is crucial for the successful recognition of music notation. It is worth observing that when performing the recognition process with traditional computer vision techniques, TIFF files can be used even though their file size is large compared to other file formats. However, attempting to perform the recognition process with techniques related to deep learning, PNG and JPEG file formats will prove more efficient due to their smaller file size especially if the dataset contains a large quantity of images.

The second category of OMR inputs is online OMR which consists of capturing real time user interactions on electronic devices such as a touch screen (Calvo-Zaragoza et al., 2019). This category goes beyond the scope of this thesis.

The third category of OMR inputs distinguishes different music notation systems and it comprises systems such as the contemporary CWMN system which is the most common, and earlier notation systems such as the neumatic notation, Guido's system of notation, the Franconian style of notation, and the squared notation (Calvo-Zaragoza et al., 2019). The recognition of historical music manuscripts may easily be ignored but it is important for the preservation of our cultural heritage and the musicological research on earlier music traditions (Fornés, 2005). Due to the different nature of historical music manuscripts, their recognition requires different methods and algorithms compared to notation made with the CWMN system (Fornés, 2009). Notation systems also include instrument specific systems such as percussion notation and tablatures for string instruments such as the guitar or the lute, and notation systems from other than Western music traditions (Calvo-Zaragoza et al., 2019). From the musicological perspective it would be interesting to have OMR end-to-end systems that could automatically examine and analyse differences among music from cultural traditions.

The fourth category of OMR inputs refers to music complexity (Calvo-Zaragoza et al., 2019). Some music scores are inherently more difficult to recognize computationally than others because the level of music complexity itself can vary greatly. However, as there are always exceptions to the rule, this does not always mean that an easy to play piece of music is always easy to recognize, or conversely that a difficult to play composition is

difficult to recognize (Byrd & Simonsen, 2015). Therefore, there are many ways to classify music complexity in relation to OMR and one way distinguishes the amount of voices in a composition, as such music complexity consists of the following four categories: (1) monophonic, (2) homophonic, (3) polyphonic, and (4) pianoform (Calvo-Zaragoza et al., 2019).

The simplest category is monophonic music which consists of one single voice played or sung at a time (Calvo-Zaragoza et al., 2019). This means that it includes only one melody line with no accompaniment. Monophonic music is relatively easy to recognize, thus most of the challenges are related to the recognition of multi-voiced notation (Byrd, 1984). However, as illustrated in Figure 23, the level of complexity of monophonic music varies depending on the qualities of its induvial music symbols and their relationship to other symbols. For instance, Paganini's 24 Caprices are extremely difficult to play on the violin, but the excerpt presented in Figure 23 is not so difficult to recognize.

Figure 23. Monophonic notation (Paganini, 1802-17).

Homophonic music consists of multiple voices which are played at the same time to form a chord that is played as a single voice (Calvo-Zaragoza et al., 2019). On the other hand, polyphonic music refers to multiple voices appearing on one single stave. This form of multi-voiced music is distinctive of the baroque and renaissance periods (Taruskin, 2006).

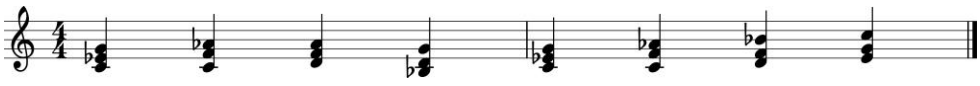

Figure 24. A homophonic chord progression.

From the perspective of OMR, the complexity related to multiple voices on a single stave is partly due to the graphical properties of its individual symbols; the stems for the upper voices point up and the stems for the lower voices point down, and additional symbols such as slurs and ornaments are usually placed outside the stave (Byrd, 1984). Sometimes voices can also cross each other so that a higher voice is momentarily lower on

the stave and vice versa (Byrd & Simonsen, 2015). Additionally, overlapping music symbols are commonly found on these types of scores.

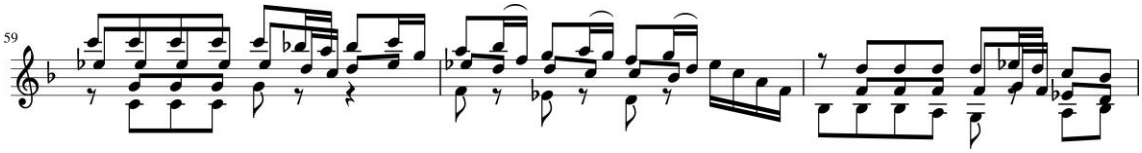

Figure 25. Polyphonic notation (Bach, 1720).

The level of difficulty for OMR increases with scores where there are numerous staves from which some staves have two or more voices, and the most complex scores are those that additionally involve voices that temporarily cross from one stave to another (Byrd, 1984). Therefore, Byrd and Simonsen (2015) also distinguish the pianoform category which refers to scores with multiple staves where multiple voices on single staves interact with other staves (Calvo-Zaragoza et al., 2019). In addition to multiple voices, piano scores must convey the piano's damper pedal and the sostenuto pedal. Sometimes they may need to be depicted with more than two staves (Gold, 2014).

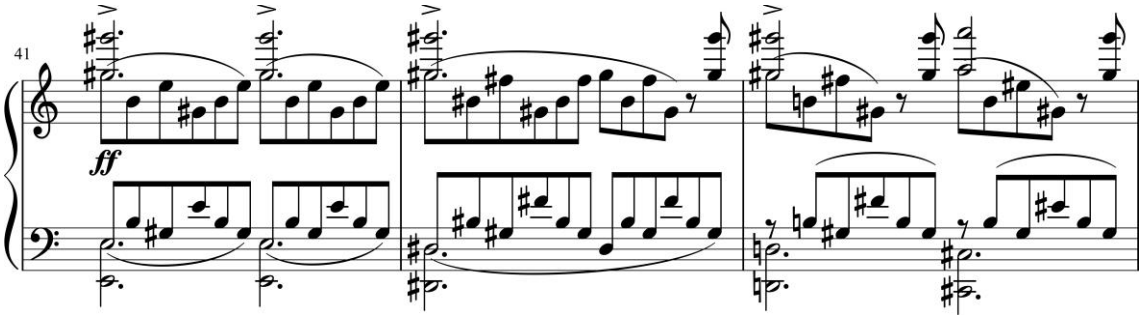

Figure 26. Multiple staves with multiple voices on each (Liszt, 1850).

Byrd (2018a) has collected an interesting and extensive list of music scores that contain examples of exceptionally complex notation published by well-known mainstream composers such as Ravel, Scriabin, Bartok, Straus, and Verdi, just to mention a few. The basic idea is to challenge the thought that CWMN could be taken apart into smaller components and mechanical rules for the subsequent hardcoding of the rules with traditional methods of computer vision. Nonetheless, this point of view on the complexity of notation also questions the capability of deep learning to deal with all the intricacies related to

music notation. This being the case, it seems best to concentrate on the areas of computer vision and deep learning that can be researched and advanced at the moment.

The fifth category of OMR inputs refers to the quality of the input. It can vary on a range from flawless documents to extremely degraded historical manuscripts. Recently printed documents are usually flawless and the easiest to recognize for the computer. Extremely degraded music scores usually consist of old manuscripts with faded ink, stains, or bleed-throughs. How the input was acquired also affects the outcome, as such, scanned images and photographs can greatly vary in quality. (Calvo-Zaragoza et al., 2019.)

3.4 OMR outputs

Calvo-Zaragoza et al. (2019) argue that OMR lacks precise methods and metrics for the evaluation of OMR outputs. As the purpose of the recognition defines the output, they suggest an evaluation system that is based on the categorization of the music comprehension level needed to recognize the input. It is worth observing that the computational level of difficulty can be high even in the lowest categories of OMR outputs. They propose four categories which are (1) document meta-data extraction, (2) search, (3) replayability, and (4) structured encoding. As illustrated in Figure 27, the meta-data extraction category requires the least amount of understanding on music notation and the structured encoding category requires the highest level of knowledge on music.

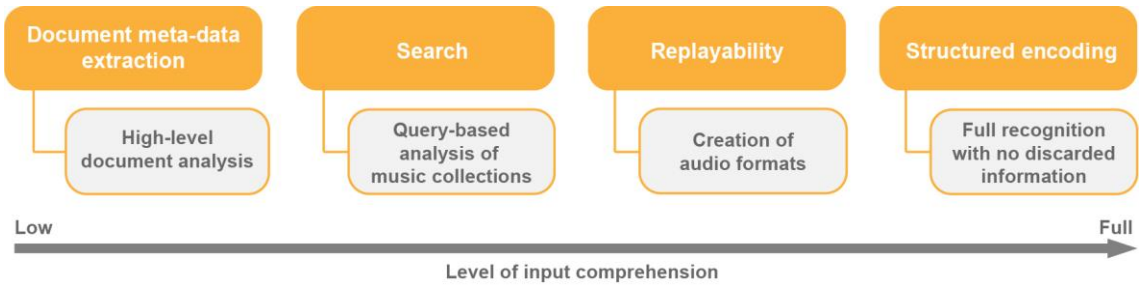

Figure 27. OMR output categories categorized by input comprehension level (Calvo-Zaragoza et al., 2019).

The document meta-data extraction category of OMR outputs refers to use-cases that involve only a low-level understanding of the music score to be recognized. Useful applications of this category are, for example, the search of music scores among other

types of images, the identification of notation systems, the estimation of the era a score was written in, the identification of the writer or composer, and the identification of the number of instruments found in a score (Calvo-Zaragoza et al., 2019).

Even though a vast amount of public domain music scores is becoming available in virtual libraries, there still is a substantial number of images that incorporate scores together with other images and text and that have not been researched or made available to the public. Bainbridge and Bell (2006) have used methods of computer vision such as the Hough transform and the run-length ratios to create search engine filters for the identification of scores in images. In the case of music scores, the objects to be identified for the classification are staves and bar lines. The run-length ratio method measures the ratio of adjacent black and white pixels from vertical scans on an image with the objective of finding staves on an image. The run-length ratio filter has been found to be fast and works well even with small-sized images. As seen in the previous chapter, subsequent research on this area has involved deep convolutional neural networks which has also meant successfully identifying music notation from other images (Pacha, 2019).

The search category of OMR outputs refers to use cases that involve query-based analysis of collections of music scores from databases such as IMSLP, DIAMM, and CPDL (Calvo-Zaragoza et al., 2019). The IMSLP, also known as the Petrucci Music Library, is one of the largest virtual music libraries of public-domain music with over 500,000 published music scores and over 60,000 recordings from almost 20,000 composers (IMSLP, 2019). The DIAMM is a virtual image library focused on medieval and early modern polyphonic music manuscripts up until the year 1600 with over 60,000 images and almost 4,000 manuscripts (DIAMM, 2019). The CPDL is, in turn, a virtual music library focused on choral and vocal public domain music with 10,000 scores (CDPL, 2019). It is important to notice that music scores are the intellectual property of the composer and the publisher defines the permitted use of the notation. Thus, most public databases only share notation the copyright of which has expired and thus can be freely distributed (Calvo-Zaragoza et al., 2019). For example, if the author of a composition died before 1950 and the composition was first published before 1925, the composition is always public domain in Canada, the United States, the EU, and Russia (IMSLP, 2019).

The search category involves a deeper level of understanding of notation than the document meta-data extraction category. As opposed to the queries in the document meta-

data extraction category, the search category queries are involved with music semantics such as melodic sequences, chord progressions, as well as interval structures from specific compositions, measures, or even delimited pixel areas (Calvo-Zaragoza et al., 2019). Most libraries still provide music scores as simple scanned images or PDF files which therefore require further treatment in order to transform them into encoded OMR outputs. Libraries and communities in charge of digitizing music collections have a growing need to automate the process of music notation recognition because more than often scanning and metadata entering are done manually which is time consuming, expensive and prone to errors (Pacha, 2019).

The replayability category of OMR outputs refer to OMR use-cases that create audio-file formats from music scores. This category requires a higher level of music comprehension than the search category. The parameters that need to be recognized from the score are pitch, tempo, rhythm, and dynamics, and depending on the purpose of the output, other information presented on the score can possibly be disregarded. Hence, human intervention is not necessarily needed in all cases for the conversion of the input data into audio format (Calvo-Zaragoza et al., 2019). The most common audio-file format is MIDI, which is a standard protocol that connects musical devices such as digital instruments, computers, tablets, and smartphones for recording, playing, and editing music by sending instructions from one device to another (MIDI Association, 2019). Given that the majority of music still remains in paper-format and has probably never been recorded, OMR that would automatically generate MIDI files from historical manuscripts would open up new possibilities in quantitative musicological analysis. Additionally, these audio files would provide missing accompaniment for practising musicians as well as facilitate the understanding of students of more complex music notation. (Calvo-Zaragoza et al., 2019.)

The structured encoding category of OMR outputs refers to use-cases that involve the full recognition and comprehension of a music score. This is the most challenging category because no information should be overlooked or disregarded during the recognition process. Unfortunately, at the moment there is no standard computer-readable format able to deal with all the information that might be presented on a music score. MusicXML and MEI formats are the best available options, but they still need improvement as they cannot manage all conditions that can be encountered in music scores. However, both formats are continuously being developed. (Calvo-Zaragoza et al., 2019.)

MusicXML is a standard open format music interface language based on XML developed to represent CWMN. Its main objective is the exchanging of sheet music between applications intended for music notation, analysis and archival. The current version 3.1 was released in 2017, it supports over 240 applications and was created by the W3C Music Notation Community Group (MusicXML, 2019). MEI is an open source initiative that aims at creating improved definitions for the encoding of music scores into computer-readable format. Its definitions are structured in the MEI schema which is a set of instructions for the encoding of music notation expressed in an XML schema. MEI and MusicXML are similar in the sense that they both encode music notation and express it in XML. However, MusicXML has been explicitly developed for the exchange of music information between applications, whereas MEI in addition to supporting the exchange of information also encodes musical intellectual content. Currently MEI supports CWMN, neume notation from the Middle Ages, mensural notation from the renaissance, string tablature, lyrics encoding, and harmony analysis (Music encoding initiative, 2019).

The main challenges that the structured encoding category face are directly related to the fact that at the moment there does not exist a powerful enough encoding system for the full representation of music notation in all possible situations (Calvo-Zaragoza et al., 2019). In order to work properly, the system should even support syntactically incorrect scores as composers do not always follow all the rules of music in their work. Additionally, it is worth remembering that music is performed by interpretation which means that a composition is often studied by a musician for a long period of time before being performed. This study may include tasks such as learning about the composer's intentions, understanding the composition's mood and feel, and acknowledging the limits of the instruments to be used (Bainbridge and Bell, 2001). Thus, having a computer imitating this entire human process has had only very limited success. For this reason, research on OMR has started focusing on the idea of making computers learn music notation, theory, and interpretation with deep learning approaches (Pacha, 2019).

OMR systems that attempt to perform structured encoding include software such as SmartScore, Capella-Scan and PhotoScore (Rebelo et al., 2012). There is also an open-source application called Audiveris. The table below lists the most relevant OMR software available at the moment.

Table 1. The most relevant OMR software listed by their inputs and outputs.

Software	**Binarized OMR Input**	**OMR Output**
SmartScore	PDF, TIFF	Finale, MIDI, MusicXML, NIFF, MP3, PDF
SharpEye	PDF, TIFF, BMP	MIDI, MusicXML, NIFF
PhotoScore	PDF, BMP	MIDI, MusicXML, NIFF, WAV/AIFF
Capella-Scan	PDF, TIFF, BMP, GIF, PNG, PS	Capella, MIDI, MusicXML
VivaldiScan	TIFF, BMP	NIFF, MIDI, MusicXML
Audiveris	PDF, TIFF, BMP, GIF, JPG, PNG	PDF, MusicXML
Gamera	TIFF, PNG	MIDI, XML, GUIDO

Due to the lack of rigorous standards related to OMR outputs as well as music recognition methods, the systematic comparison of OMR software is challenging. A proper evaluation method should be public, systematically defined and verified, and it should provide meaningful results that advance research in OMR (Calvo-Zaragoza et al., 2019). Hence, the systematic comparison of OMR software is outside the scope of this thesis.

4 Optical music recognition architecture

4.1 The general framework

OMR has been researched since the 1960s and a general framework has been established and adopted by several authors for the definition and decomposition of problems related to OMR. The general framework aims at solving most OMR problems with traditional methods of computer vision and pattern recognition, but some newer deep learning methods are currently being integrated into the model. The framework consists of four phases which are (1) image pre-processing, (2) music symbol recognition, (3) music notation reconstruction and (4) final representation construction as indicated in Figure 28. The main objective of the framework is to provide the technical means for the recognition of a music score, its subsequent analysis and interpretation, as well as its storage in machine readable format. (Rebelo et al., 2012.)

The purpose of this chapter is to provide an overview of relevant computer vision methods and algorithms available for OMR. The purpose is not, however, to provide an extensive study of all existing approaches and their intricacies. The idea is more about summarizing the most important technical problems and considering possible ways of solving some of them especially with approaches related to deep learning.

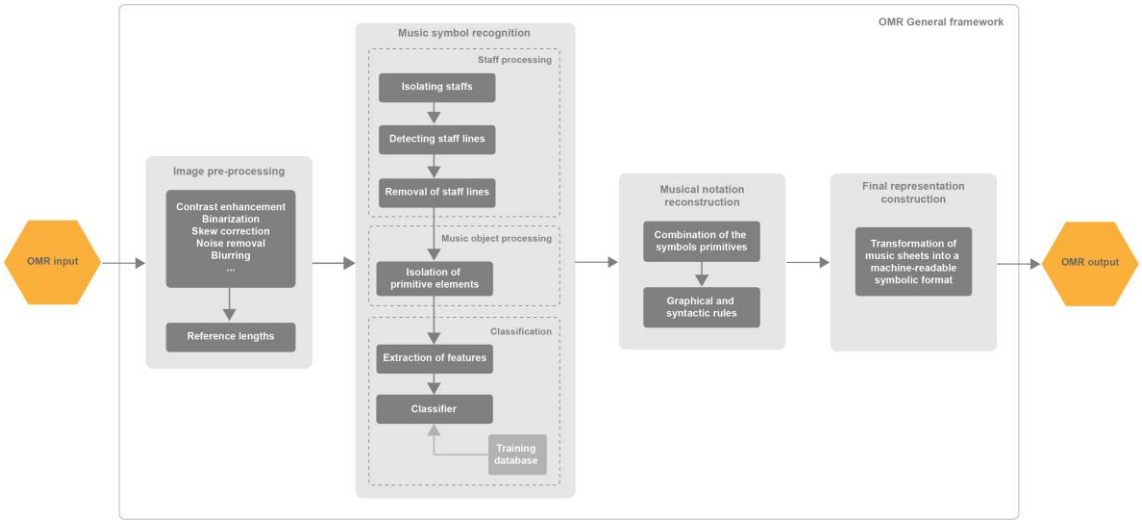

Figure 28. OMR General framework (Rebelo et al.,2012)

4.1.1 Image pre-processing

The basic idea of the image pre-processing phase is to enhance the original input as much as possible by using different image editing techniques such as binarization, contrast improvement, noise removal, skew correction, and blurring (Rebelo et al., 2012).

Binarization is the first task to be carried out and its objective is to transform the pixel image into a black and white or binary image and determine which artifacts are part of music notation and which ones are not (Novotný & Pokorný, 2015). All irrelevant parts such as background noise, steins, and ink bleed-through ought to be disregarded to achieve higher computational efficiency (Rebelo et. al, 2012). Noise refers to artifacts in an image that prevent the efficient recognition of an image. Noise comes in many forms, the most common being random noise which appears as a grainy image texture, salt and pepper noise which is encountered in the form of unwanted black and white pixels, and coloured noise which results from arbitrary oscillations in the frequencies of an image (Kinser, 2019). Thus, binarization does not require any knowledge of music. Its objective is the reduction of the amount of information to be processed (Rebelo et al., 2012).

The most relevant algorithms used for binarization are the global thresholding method and the adaptative thresholding method (Rebelo et al., 2012). In computer vision, image thresholding refers to changing the colour value of a pixel to either black or white according to a threshold value (Kapur, 2017). Global image thresholding is especially suitable for images with a uniform background and in this case a global threshold value can be applied for the entire image, which enables the algorithm to perform at high speed (Novotný & Pokorný, 2015). Nonetheless, there are frequently situations when the image background is non-uniform. This situation is particularly encountered when working with older music manuscripts and low-quality music documents (Rebelo et al., 2012). The effective binarization of images with a non-uniform background requires calculating different threshold values for different parts of the image and the process is referred to as adaptative thresholding (Kapur, 2017). The disadvantage of the adaptative method is that it requires a longer processing time than the global method (Novotný & Pokorný, 2015). Another common approach to binarization is the Niblack method which uses the local mean and standard deviation of the nearest pixel's intensity values for setting the threshold (Rebelo et al., 2012). The binarization of images unfortunately introduces noise into the image which can be a problem especially if the original image is of low quality. Although it will affect efficiency, this problem could be solved by making higher quality

scans from the original OMR input. It is also worth observing that often it is assumed that music scores are grayscale or black and white images, but the application of colour information in the analysis of the image could prove useful, for example, in the context of pre-printed staff-paper where staff-lines and hand-written music symbols usually appear in a slightly different colour (Novotný & Pokorný, 2015). Historical music manuscripts also use colours in music scores as seen, for example, in Guido's notation system and Philippe de Vitry's mensural form of notation (Kelly, 2019).

The image pre-processing phase also includes the calculation of staff-line and staff-space heights which provide a baseline for the comparison of music symbol sizes (Rebelo et al., 2012). In essence, the size of music symbols used in the score must be identified before starting the recognition of individual symbols (Bainbridge & Bell, 2001). Some authors refer to this task as the initiation of segmentation and it is commonly based either on RLE or on the horizontal projections method. In RLE, black runs calculate the staff-line height and white runs the staff-space height (Novotný and Pokorný, 2015). The calculation process assumes that staff-lines cover the largest part of the music score and that their height is the smallest shape height found on the score (Bainbridge & Bell, 2001). Any vertically positioned black run which is twice the height of a staff-line will be removed as well as any object that has a width less than the height of the staff-space (Rebelo et al., 2012). The horizontal projections method involves the mapping of a two-dimensional bitmap into a one-dimensional histogram by calculating the number of black pixels in each row of the bitmap (Bainbridge & Bell, 2001). As a result, staff-lines appear as peaks in the histogram. Horizontal projections are often performed in different angles in order to deal with distorted staff-lines (Novotný & Pokorný, 2015). Another model for the identification of staff-lines is the application of vertical scan lines which are based on line adjacency graphs. This algorithm calculates the sum of two consecutive vertical runs and selects the ones that appear most often (Rebelo et al., 2012). Staff-line height recognition problems are often encountered especially in handwritten music scores where staff-lines are not completely parallel, or in printed scores where staff-lines appear distorted due to low quality digitization (Rebelo et al., 2012).

4.1.2 Music symbol recognition

Music symbol recognition is the second phase of the general framework and it consists of three main activities which are (1) staff processing, (2) music object processing, and

(3) music object classification (Rebelo et al., 2012). These three activities are further divided into sub-tasks as described below.

The staff processing activity includes subtasks that are staff-line isolation, staff-line detection, and the possible removal of the entire stave with the purpose of getting an image containing only individual music symbols (Rebelo et. al, 2012). The idea behind having isolated music symbols without the stave is the better recognition of the notation itself. Technically, the removal of staff-lines is usually carried out by simply replacing identified black staff-line pixels with white pixels (Novotný & Pokorný, 2015). Typically, black pixels located at a distance of two pixels from the staff-line are assumed to be music symbols and therefore are not removed (Bainbridge & Bell, 2001). Sometimes, also music symbols such slurs and dynamic marking are removed with the same method (Rebelo et. al, 2012). However, the removal of staff-lines from the binary file is prone to introduce new artifacts in the image. These artifacts can for example be encountered as noise or random pixel clusters. The staff-line removal may also break the features of some music symbols or leave pixel clusters on the image if some staff-line shapes are erroneously detected as music symbols. For these reasons, some authors prefer ignoring the stave instead having it removed from the file (Novotný & Pokorný, 2015).

The previously described staff processing activities are followed by music object processing and music object classification (Rebelo et al., 2012). These activities are also referred to as music symbol segmentation (Novotný & Pokorný, 2015). In computer vision, segmentation refers to the process of splitting an image into its smaller parts in order to understand its content (Kapur, 2017). Similarly, in OMR the main purpose of segmentation is the locating, identification, and classification of individual music symbols within the score. The intrinsic complexities related to music notation make these activities particularly complicated. Especially notation with superimposed and touching symbols pose a challenge as they may easily result in non-identifiable artifacts on the image (Bainbridge & Bell, 2001). Low quality printing and scanning as well as paper degradation pose a big challenge to this stage of process (Rebelo et al., 2012).

The music object processing activity is based on the isolation of individual music primitives by decomposing the score into separate areas. Music scores are usually divided into parts that encompass a single staff (Rebelo et. al, 2012). A common method used

for the isolation of music symbols is the hierarchical decomposition algorithm which consists of the analysis, isolation, and extraction of basic music symbols such as noteheads, stems, flags, and rests (Novotný & Pokorný, 2015). It is worth noting that during music object processing, noteheads, stems, and flags are technically considered as separate objects. Thus, at this stage of the recognition process they are not assigned any musical meaning (Rebelo et. al, 2012). Depending on the used methods, the segmentation of music objects is, however, often performed simultaneously with the classification of music objects. Hence, these stages cannot always be explicitly considered separate stages of the recognition process (Novotný & Pokorný, 2015). The advantage of simultaneous segmentation and classification is that this eliminates the need for keeping track of already segmented objects that are waiting for the classification step (Rebelo et. al, 2012).

In vocal music, lyrics are also detected at this stage of the process which adds an extra layer of complexity to the recognition. After lyrics have been detected and identified, they also need to be linked back to their corresponding notes on the score also taking into account their proper syllabification (Rebelo et al., 2012). Lyrics are more than often presented on music scores in rather unpredictable locations and with inconsistent syllabification which makes the use of OCR methods ineffective (Fornés, 2005). Traditional OCR is based on language models that include word dictionaries but the need for syllabicated dictionaries decreases the amount of languages available and increases the amount of unidentified inflections (Burgoyne et al., 2009). Baselines are often calculated for lyrics and their corresponding notes by using common computer vision methods such as local horizontal projections and vertical run-length encoding (Rebelo et al., 2012).

The music object classification step aims at recognizing each individual music symbol by the extraction of their musical features (Rebelo et al., 2012). Classification is frequently started by the recognition of the simplest primitives, such as noteheads, stems and accidentals by using a plethora of pattern recognition techniques such as template matching, the Hough transform method, the line adjacency graph method, the character profile method, and horizontal projections (Bainbridge & Bell, 2001).

Template matching is a method used in computer vision for finding the coordinates of specific parts of an image by matching a template image to the image to be analysed (Rebelo et al., 2012). The Hough transform refers to a general framework that is given parametrized equations of shapes such as straight lines, curves, circles or ellipses as

input and then it identifies these shapes in an image (Kapur, 2017). The Hough transform can easily locate lines of different thickness, orientation, and length (Kinser, 2019). However, other approaches such as the line adjacency graph method is preferred for the detection of curved lines (Rebelo et al., 2012). The disadvantage of the Hough transform method is that imperfections on the thickness of a line often result in overlapping curves in the image. Therefore, the Hough transform method is suitable for recently printed high quality scores. The character profile method which measures the perpendicular distance of the object's outline to reference the axis is used for the identification of accidentals, rests, and clefs (Rebelo et al., 2012). Music objects can also be detected by the application of projections using features extracted from projection profiles (Novotný & Pokorný, 2015).

The music object classification is also the stage of the recognition process where deep learning algorithms are becoming increasingly popular and where a clear shift can be noticed from traditional computer vision methods to deep learning approaches. It must be emphasized, however, that the integration of machine learning and deep learning into OMR is still in its early stages and there are no solutions yet available to solve all open matters. Some of the approaches that involve deep learning include methods such as the k-nearest neighbour rule and the hidden Markov models. The k-nearest neighbour rule is used for classification, and the hidden Markov model is used as the classifier by labelling a set of music symbols. The set is divided into training sets and test sets from which the model will learn from (Rebelo et. al, 2012). The k-nearest neighbour classifier is one of the simplest methods used for classification. It simply compares the object in question to the labelled objects found on the training set. The k-nearest neighbour classifier is efficient, but its disadvantage is that it needs the entire training set which can slow down the recognition process when using larger training sets (Solem, 2019).

As deep learning has fostered advances in OMR, several public datasets have been made available to facilitate the classification of music symbols. These datasets include HOMUS, the Universal Music Symbol Collection, the CVC-MUSCIMA, MUSIMA++, DeepScores, PrIMuS and Camera-PrIMuS. Special open source tools such as MUSICMarker and MuRET have also been developed for the creation of datasets.

The Universal Music Symbol dataset is a collection of several datasets that have been combined into a collection of 90,000 music symbols from which 74,000 symbols are

handwritten and 16,000 are printed (Pacha & Eidenberg, 2017). The CVC-MUSCIMA is a dataset designed for writer identification that has 1,000 sheets of handwritten music scores written by 50 different musicians (Fornés et. al, 2012). The MUSCIMA++ is a dataset designed for handwritten music symbol detection which has over 91,000 music symbols of basic music primitives and higher-lever notation objects such as key signatures and time signatures (Hajič & Pecina, 2017). DeepScores is a dataset designed for music object classification, semantic segmentation, and object detection that has 300,000 images of printed music sheets obtained from MuseScore (Tuggener et. al, 2018). The PrIMuS dataset has been designed for the identification of melodies and it has over 87,000 sequences of melodies saved as PNG images, and in the MIDI and MEI format (Calvo-Zaragoza & Rizo, 2018). The Camera-PrIMuS dataset contains the same sequences of melodies as the PrIMuS dataset, but the PNG images have been distorted in order to simulate real life situations (Calvo-Zaragoza & Rizo, 2018).

The creation of a datasets for training purposes in deep learning is expensive and time consuming because of the large amount of data needed for the datasets. The MUSCIMA++ dataset was created with a tool called MUSICMarker which is an open source application developed in Python that is flexible enough to be applicable for the creation of datasets for similar purposes (Hajič & Pecina, 2017). MuRET is also an open source tool for music recognition, encoding and transcription developed with Angular and Spring Boot which can be used in all phases of the recognition process, including manuscript source to encoded OMR output (Calvo-Zaragoza et al., 2018).

4.1.3 Music notation reconstruction and final representation

The music notation reconstruction phase consists of two activities which are the (1) the combination of music symbol primitives and (2) the application of graphical and syntactic rules to the file in order to reproduce its musical meaning. Therefore, this phase encompasses the application of music syntax and semantics to recognise music symbols (Rebelo et. al, 2012). The activities involved are particularly challenging due to music notation's two dimensionality which, as seen earlier, means that the meaning of music symbols depends on their horizontal and vertical location on the stave as well as their relationship to other music symbols (Novotný & Pokorný, 2015).

The combination of music symbol primitives is performed by interpreting the relationships between detected music objects. This has traditionally been performed by hardcoded rules that address certain parts and circumstances of music notation (Calvo-Zaragoza et al., 2019). These rules are basically related to tonality, accidentals, and tempo which means that the identification of the key signature and accidentals is crucial for the correct recognition of the music score (Rebelo et al., 2012). A common method for this step of the process is the formation of musical features based on the application of music knowledge grammars that specify and add meaning to the relationships among music symbols (Novotný & Pokorný, 2015). These grammars determine the way music objects must be processed, the way musical events should be made, and the way music objects should be segmented. A common grammar has been implemented with *λ*Prolog also known as lambda Prolog which is a logic programming language that has semantic attributes connected to C libraries for pattern recognition and decomposition (*λ*Prolog, 2020). The grammar is developed using DCG techniques of parsing at a graphical and at a syntactic level (Rebelo et al., 2012). The parser structure consists of a list of segmented and non-labelled music objects that are connected to music grammars. The process starts with the labelling of objects and the detection of errors according to the available grammar. Another common approach for the combination of music objects and grammar rules also uses CDG techniques to specify the relationships between musical symbols, but it also uses a system called CANTOR that allows the user to manually define the rules for the notation (Rebelo et al., 2012). The disadvantage is that this method is time consuming and prone to errors although it leaves room for the user to make decisions on needed rules.

The last phase of the framework is called final representation construction and it consists of activities related to the transformation of music scores into machine readable format (Rebelo et al., 2012). The techniques used in this phase are greatly determined by the OMR output format which can be, for example, MIDI, MusicXML, MEI (Bainbridge & Bell, 2001). The most significant OMR output formats are described in chapter 3.4.

A considerable amount of research in OMR has been carried out and many competing methods and algorithms have been created to solve OMR's inherent problems. Even though advances have been made in computer vision, most approaches often focus on very specific and well-formulated problems that cannot easily be extended to slightly different circumstances which is a prerequisite of an OMR system. The outcome is that

the selected methods frequently disregard or misinterpret some information presented on the music score, which easily results in errors that spread from one phase to the next within the framework and eventually lead to an incorrect OMR output. During the past years, several methods and algorithms have improved individual steps within the framework, but currently there is still no system that can automatically recognize a large set of music notation accurately and effectively and without human intervention. (Pacha, 2019.)

4.2 Updates on the general framework

The general framework forms the foundation for the technical understanding of OMR. However, it will undoubtedly need updating as new developments emerge in the field of computer science and particularly deep learning (Bainbridge & Bell, 2001). Research has concluded that many problems found in OMR can be addressed as machine learning problems, which can be solved with deep learning (Pacha, 2019). After having introduced machine learning into OMR, the technical problems related to recognition have been slightly reformulated; however, the general framework remains valid as open issues are still related to image pre-processing, object detection, semantic reconstruction, and encoding. Recently introduced techniques of deep learning have been able to solve some open issues of image pre-processing and music object detection (Pacha, 2019). However, semantic reconstruction and encoding remain a complete challenge. It also has to be taken into account, that recently introduced techniques may not always be able to deal with all kinds of situations until they have matured and knowledge on their limitations and proper application has spread within the community.

Deep learning has been successful in music object detection. For instance, deep learning models with convolutional neural networks are now able to recognize rather well music objects without staff-line removal (Calvo-Zaragoza et al., 2019). Therefore, the staff processing step and the music object detection step on the general framework will highly likely become obsolete at some point in the future. It is worth emphasizing, however, that breakthroughs made in other areas of research such as image classification, speech recognition and handwriting recognition are based on neural networks that are only able to deal with one-dimensional outputs such as sequences of words (Pacha, 2019). Hence, music notation's two-dimensionality makes it difficult to train optimized models for OMR. Currently the recognition process is limited to certain types of music notation such as the

recognition of mensural notation with Markov models, and the recognition of printed and handwritten monophonic notation with deep neural networks. At present, there is no suitable model for more complex music notation such as polyphonic music or grand-piano form. Deep learning algorithms are also often based on statistical models that provide probabilities over established hypothesis. The determination of musical syntax and semantics by using statistical models could also prove to be an area worth researching (Calvo-Zaragoza et al., 2019).

Thus, traditionally used techniques of computer vision are not to be disregarded in OMR, and even though advances have been made in deep learning, a lot of research is still needed in order to implement the techniques successfully in the recognition process. This means that the transition from traditional techniques to new ones cannot be done abruptly and the suitability of each individual method has to be evaluated on a case by case basis.

5 Discussion and conclusion

The objective of this thesis is to provide an overview of OMR and address the challenges and possibilities related to it. OMR stands for optical music recognition and in the scientific community it has been defined as “the field of research that investigates how to computationally read music in documents” (Calvo-Zaragoza et.al, 2019). OMR is closely related to other fields of computer science such as computer vision, machine learning and deep learning. Altogether, it is also closely related to musicology and music information retrieval. It is worth observing that OMR does not, however, advance any of these fields, but it uses the knowledge they provide. Nonetheless, OMR focuses on specifying what kind of information can be retrieved from music notation, how the retrieval is to be designed and executed, and what the constraints related to different forms of notation are.

Advances in the field of OMR contribute especially to the preservation of cultural heritage, music education, music composition and practice, as well as research in musicology. In practice, OMR facilitates, for instance, the creation of large-scale searchable music databases for musicological analysis, the publication of archived music scores, and especially the development of software for the automatic recognition of printed and handwritten music notation as well as the encoding of the output into an appropriate format such as MusicXML, MIDI or MEI.

OMR has been researched for decades but still no computer system is able to overcome all the challenges related to music recognition. Research in OMR is confronted with challenges that are directly related to the complexity of music notation and the lack of effective methods and algorithms capable of dealing with these problems. Music notation has evolved over the centuries into a sophisticated visual language that has its own vocabulary, syntax, and semantics, and as any other language it also has its own practices, dialects, and styles. Furthermore, developments in music continuously introduce new forms of expression. Music notation also encompasses a vast amount of music symbols that are to be interpreted differently depending on the context and circumstance they are presented in as well as their relationship to other symbols. Notation is also two-dimensional which means that the assigned meaning depends on their position on the vertical and horizontal axis of the stave. The two-dimensional nature of music notation is one of

the features that poses the greatest challenge to the recognition process as at the moment there are no techniques, neither in computer vision nor in deep learning, that can deal with it appropriately, accurately and efficiently. Additionally, music is sometimes presented on scores ambiguously, at times the composer breaks rules of music deliberately, and every so often information has been disregarded either intentionally or unintentionally. Research in OMR suggests that this is also where the boundaries for OMR are to be established as the computer cannot complete missing information that requires a sound knowledge of music theory and music history as well as the ability to interpret music intuitively. At present, machine learning is uncapable of dealing with such an effort.

From this perspective OMR should focus on investigating how deep learning could further improve music object detection, for example, with convolutional neural networks so that all symbols available in SMuFL could be properly be detected and identified. As rather large music symbol and notation datasets already exist, it could be valuable to try to transfer already trained networks among different datasets. Nonetheless, the mere detection of music objects does not suffice the requirements of advanced OMR systems which should also be able to assign correct musical semantics to music objects. However, the semantic reconstruction should be attempted only after music object detection has been successfully achieved. Rules cannot simply be applied to objects unless they have first been identified. As deep learning algorithms are often based on statistical models that provide a probability over a set of hypotheses, it could be valuable to investigate the possibility of using probabilistic models for the semantic reconstruction of music scores. This is an area that is still rather unexplored in OMR. Thus, an advanced end-to-end OMR system should be flexible enough to let the user verify the outcome of the recognition and allow the user to change the interpretation when needed. On the other hand, the application of probabilistic models could altogether bring completely new perspectives to music analysis and composition. It is also worth observing that there are large amounts of archived music manuscripts that have never been investigated and made available to public. It would be truly interesting to analyse these scores with advanced methods of machine learning.

All in all, an overall OMR is still to be considered a problem to be solved, but advances in machine learning together with computer vision are slowly taking it further. Hopefully in the years to come, a system that satisfies the requirements of CWMN is successfully completed and made available to the public.

References

Bach, Johan Sebastian. 1720. Solo violin sonata No.1 in G minor. Open music source from IMSLP Petrucci Music Library.

Bainbridge, David; Bell, Tim. 2001. The challenge of Optical Music Recognition. Computers and the Humanities. URL: https://www.researchgate.net/publication/220147775_The_Challenge_of_Optical_Music_Recognition. Accessed 17 March 2020.

Bainbridge, David; Bell, Tim. 2006. Identifying music documents in a collection of images. ResearchGate database. URL: https://www.researchgate.net/publication/29486838_Identifying_music_documents_in_a_collection_of_images. Accessed 20 April 2020.

Burgoyne, John; Devaney, Johanna; Ouyang, Yue; Pugin, Laurent; Himmelman, Tristan; Fujinaga, Ichiro. 2009. Lyric extraction and recognition on digital images of early music sources. ISMIR. ResearchGate database. URL: https://www.researchgate.net/publication/277297348_Lyric_extraction_and_recognition_on_digital_images_of_early_music_sources_ISMIR. Accessed 17 January 2020.

Byrd, Donald Alvin. 1984. Music notation by computer. PhD. Diss. Computer Science Department, Indiana University.

Byrd, Donald; Simonsen, Jakob Grue. 2015. Towards a standard testbed for optical music recognition: definitions, metrics and page images. ResearchGate database. URL: https://www.researchgate.net/publication/282403912_Towards_a_Standard_Testbed_for_Optical_Music_Recognition_Definitions_Metrics_and_Page_Images. Accessed 15 June 2020.

Byrd, Donald. 2018a. Extremes of conventional music notation. Website. URL: http://homes.sice.indiana.edu/donbyrd/CMNExtremes.htm. Accessed 12 December 2019.

Byrd, Donald. 2018b. More counterexamples in conventional music notations. URL: http://homes.sice.indiana.edu/donbyrd/MoreCMNCounterexamples.htm. Accessed 12 December 2019.

Calvo-Zaragoza, Jorge; Rizo, David. 2018. Camera-PrIMuS: Neural end-to-end optical music recognition on realistic monophonic scores. MDPI Open access journals database. URL: https://www.mdpi.com/2076-3417/8/4/606#cite. Accessed 14 September 2020.

Calvo-Zaragoza, Jorge; Rizo, David. 2018. End-to-end neural optical music Recognition of Monophonic Scores. ismr publication database. URL: http://ismir2018.ircam.fr/doc/pdfs/33_Paper.pdf. Accessed 19 July 2020.

Calvo-Zaragoza, Jorge; Rizo, David; Inesta, José. 2019. MuRET:a music recognition , encoding and transcription tool. ACM Digital library. URL: https://dl.acm.org/doi/10.1145/3273024.3273029. Accessed 11 January 2020.

Calvo-Zaragoza, Jorge; Rizo, David; Hajič, Jan; Pacha, Alexander. 2019. Understanding Optical Music Recognition. Publication database of Cornell University. URL: https://arxiv.org/pdf/1908.03608.pdf. Accessed 14 July 2020.

CDPL: The choral public domain library. Website. URL: http://www1.cpdl.org/. Accessed 6 November 2019.

Chopin, Frédéric. 1834-35. Ballade Op.23 No.1 in G Minor. Open music source from IMSLP Petrucci Music Library.

Chopin, Frédéric. 1829-32. Etudes Op.10. Open music source from IMSLP Petrucci Music Library.

Chopin, Frédéric. 1838-39. Preludes Op. 28. Open music source from IMSLP Petrucci Music Library.

Chollet, Francois. 2018. Deep learning with Python. Manning Publications Co. New York.

DIAMM: The digital image archive of medieval music. Web site. URL: https://www.diamm.ac.uk/. Accessed 10 November 2019.

Feist, Jonathan. 2017. Berklee contemporary music notation. Berklee Press. Boston.

Fornés, Alicia. 2009. Writer identification by a combination of graphical features in the framework of old handwritten music scores. PhD. Universitat Autònoma Barcelona. Publication database of the UAB. URL: http://www.cvc.uab.es/people/afornes/publi/PhDAliciaFornes.pdf. Accessed 23 May 2020.

Fornés, Alicia. 2005. Analysis of old handwritten musical scores. Master's thesis. Universitat Autònoma Barcelona. Publication database of the UAB. URL: http://www.cvc.uab.es/people/afornes/publi/AFornes_Master.pdf. Accessed 15 January 2020.

Fornés, Alicia; Dutta Anja; Gordo, Albert; Lladós, Josep. CVC-MUSCIMA: A ground-truth of handwritten music score images for writer identification and staff removal. International journal on document analysis and recognition, Volume 15, Issue 3, pp 243-251, 2012. (DOI: 10.1007/s1002-011-0168-2).

Gould, Elaine. 2014. Behind bars – The definite guide to music notation. Faber Music Ltd. London.

Hajič, Jan; Pecina, Pavel. 2017a. In Search of a Dataset for Handwritten Optical Music Recognition: Introducing MUSCIMA++. Publication database of IEEE Xplore. URL: https://ieeexplore.ieee.org/document/8269947. Accessed 12 December 2019.

Hajič, Jan; Pecina, Pavel. 2017b. Groundtruthing (Not Only) music notation with MUSICMarker: A practical overview. Publication database of IEEE Xplore. URL: https://ieeexplore.ieee.org/document/8270213. Accessed 15 May 2020.

HOMUS: The hand-written online music symbols dataset Website. URL: https://grfia.dlsi.ua.es/homus/. Accessed 8 July 2020.

IMSLP: The international music score library project. Website. URL: https://imslp.org/wiki/Main_Page. Accessed 5 November 2019.

Joutsenvirta, Aarre; Perkiömäki, Jari. 2014. Musiikinteoria 1. Modus Musiikki Oy. Parkano.

Kapur, Saurabh. 2017. Computer vision with Python 3. Packt Publishing. Birmingham.

Kelly, Thomas. 2015. Capturing music: the story of notation. W.W. Norton & Company. New York.

Kinser, Jason M. 2019. Image operators – Image processing in Python. CRC Press. Boca Raton.

Lambda Prolog. Website.URL: http://www.lix.polytechnique.fr/~dale/lProlog/. Accessed 15 March 2020.

Liszt, Franz. 1850. Liebestraum No.3 in A major. Open music source from IMSLP Petrucci Music Library.

MIDI Association. An introduction to MIDI. MIDI Manufacture's Association. 2019. MIDI Association publication. URL: https://www.midi.org/images/easyblog_articles/43/intro-midi.pdf. Accessed 14 February 2020.

Music encoding initiative. Web site. URL: https://music-encoding.org/. Accessed 3 November 2019.

Novotný, Jiří; Pokorný Jaroslav. 2015. Proceedings of the Dateso 2015 workshop: Introduction to optical music recognition: overview and practical challenges. Charles University. Publication database of CEUR workshop proceedings. URL: http://ceur-ws.org/Vol-1343/paper6.pdf. Accessed 5 July 2020.

Pacha, Alexander. 2019. Self-Learning Optical Music Recognition. PhD. Vienna University of Technology. Publication database of the TU Vienna. URL: https://publik.tuwien.ac.at/files/publik_281264.pdf. Accessed 4 March 2020.

Pacha, Alexander; Eidenberg Horst. 2017. Towards a universal music symbol classifier. IEEE Xplore. URL: https://ieeexplore.ieee.org/document/8270207. Accessed 18 April 2020.

Mozart, Wolfgang Amadeus. 1783. Quartet No.15 in D Minor, K. 421. Open music source from IMSLP Petrucci Music Library.

MusicXML. Website. URL: https://www.musicxml.com/. Accessed 2 November 2019.

Paganini, Niccòlò. 1802-1724. Caprices for solo violin. Open music source from IMSLP Petrucci Music Library.

Paxman, Jon. 2014. A chronology of western classical music 1600 – 2000. Omnibus Press. London

Purcell, Henry. 1702. Orpheus Britannicus - The Epicure. Open music source.

Rebelo, Ana; Fujinaga, Ichiro; Paszkiewicz, Filipe; Marcal, Andre R.S.; Guedes, Carlos; dos Santos Cardoso, Jaime. 2012. Optical music recognition: state-of-the-art and open issues. International Journal of Multimedia Information. ResearchGate database. URL: https://www.researchgate.net/publication/257806547_Optical_music_recognition_State-of-the-art_and_open_issues. Accessed 16 March 2020.

Schuman, Clara. 1841-42. Piano sonata in G minor. Open music source from IMSLP Petrucci Music Library.

Schuman, Robert. 1829-32. Toccata Op.7. Open music source from IMSLP Petrucci Music Library.

Szeliski, Richard. 2010. Computer vision: algorithms and applications. Springer. New York.

SMuFL. Website. URL: https://w3c.github.io/smufl/gitbook/. Accessed 12 December 2019.

Solem, Jan Erik. 2012. Programming computer vision with Python. O'Reilly Media. Boston

Taruskin, Richard. 2006. The Oxford history of music: Music from the earliest notations to the sixteenth century. Oxford University Press. Oxford.

Tuggener, Lukas; Elezi, Ismail; Schmidhuber, Jürgen; Pelillo, Marcello; Stadelmann, Thillo. 2018. DeepScores – A dataset for segmentation, detection and classification of tiny objects. Publication database of Cornell University. URL: https://arxiv.org/abs/1804.00525. Accessed 27 November 2019.