OPTICAL MUSIC RECOGNITION: STATE OF THE ART AND MAJOR CHALLENGES

Elona Shatri
Queen Mary University of London
e.shatri@qmul.ac.uk

György Fazekas
Queen Mary University of London
g.fazekas@qmul.ac.uk

ABSTRACT

Optical Music Recognition (OMR) is concerned with transcribing sheet music into a machine-readable format. The transcribed copy should allow musicians to compose, play and edit music by taking a picture of a music sheet. Complete transcription of sheet music would also enable more efficient archival. OMR facilitates examining sheet music statistically or searching for patterns of notations, thus helping use cases in digital musicology too. Recently, there has been a shift in OMR from using conventional computer vision techniques towards a deep learning approach. In this paper, we review relevant works in OMR, including fundamental methods and significant outcomes, and highlight different stages of the OMR pipeline. These stages often lack standard input and output representation and standardised evaluation. Therefore, comparing different approaches and evaluating the impact of different processing methods can become rather complex. This paper provides recommendations for future work, addressing some of the highlighted issues and represents a position in furthering this important field of research.

1. INTRODUCTION

Music is often described as structured notes in time. Musical notations are systems that visually communicate this definition of music. The earliest known scores date back to 1250-1200 BC in Babylonia [1]. Since then, many notation systems have emerged in different eras and different locations. Common Western Music Notation (CWMN) has become one of the most frequently used systems. This notation has evolved from the mensural music notation used before the seventeenth century. Current work in Optical Music Recognition focuses on the CWMN; nonetheless, studies are also carried out for old notations, including mensural, as shown in Table 1.

Classifying music based on its difficulty is highly subjective. Nevertheless, Byrd and Simonsen [30] in their attempt to have a standardised test-bed for OMR, name four categories based on the complexity of the score [30] (see Figure 1):

1. Monophonic: music in one staff with one note at a time;
2. Polyphonic: music in one staff with multiple voices;
3. Homophonic: music in multiple staves, but each staff is monophonic;
4. Pianoform: music in multiple staffs and multiple voices with significant structural interactions.

OMR has been researched for the last five decades; nonetheless, a unified definition of the problem is yet to emerge. However, Calvo-Zaragoza [31] offers the following definition of OMR.

Definition 1.1 *"Optical Music Recognition is a field of research that investigates how to read music notation in documents computationally."*

The importance of OMR is evident both in the abundance of sheet music in archives and libraries, much of this is yet to be digitised, and in the common practice of musicians. Paper remains the first medium authors use to write music. By taking a picture of a score, OMR would enable us to later modify, play, add missing voices and share music using ubiquitous digital technologies. It also enables search capabilities, which are especially crucial for long pieces or large catalogues in music information retrieval and digital musicology. Other advantages of OMR include conversions to different sheet music formats (e. g. Braille music notation) and the ability to archive musical heritage [32].

Fundamentally, OMR's goal is to interpret musical symbols from images of sheet music. The output would be a transcribed version of the sheet, which is also machine-readable, i.e., musical symbols can be interpreted and manipulated computationally. The usual output formats are MusicXML and MIDI. These formats will include musical attributes and information such as pitches, duration, dynamics and notes.

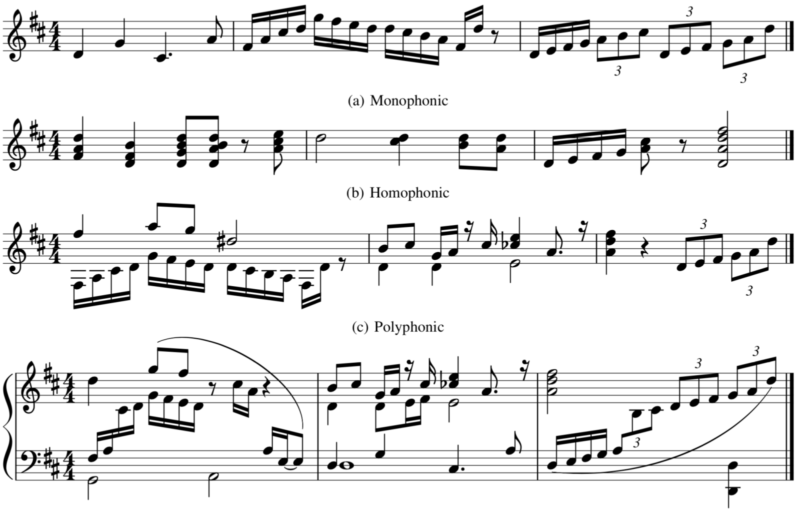
OMR has previously been referred to as Optical Character Recognition (OCR) for music. However, music scores carry information in a two-dimensional structure, ordered sequence of musical symbols and the set of most often co-occurring musical events. In contrast, OCR deals with sequences of characters and words that are one-dimensional.

The field of OMR saw its beginnings in the late 1960s with pioneers like Pruslin [33], Prerau [34] and continued with Kassler [35] reviewing the dissertations mentioned above. Subsequently, other researchers including Fujinaga [2], Carter [36], Ng [4], Bainbridge and Bell [10] joined the field.

More recently, the success deep learning has had in improving text and speech recognition has triggered a

References	CWMN	Old	Typeset	Handwritten
Fujinaga [2], Coüasnon et al. [3], Ng and Boyle [4], Chen et al. [5], Vidal [6], Bui et al. [7], Huang et al. [8]	✓		✓	
Ng et al. [9], Bainbridge and Bell [10], Gocke [11], Rebelo et al. [12], Fornés et al.[13], Pinto et al. [14], Hajič and Pecina [15], Roy et al. [16], Pacha et al. [17], Tuggener et al. [18], Baró et al. [19, 20]	✓			✓
Calvo-Zaragoza and Rizo [21], Wen et al. [22], Pacha and Eidenberger [23, 24], Calvo-Zaragoza et al. [25]	✓		✓	✓
Calvo-Zaragoza et al. [26, 27], Huang et al. [28], Tardón et al. [29]		✓		✓

Table 1: Studies conducted in CWMN (Common Western Music Notation), and old notations (used before the CWMN, mostly mensural notations


(a) Monophonic

(b) Homophonic

(c) Polyphonic

(d) Pianoform

Figure 1: A visual representation of the four categories of music notations [30, 31]

paradigm shift in OMR as well. One of the most comprehensive reviews on OMR was written in 2012 by Rebelo et al. [37]. However, at that time, the field had not yet seen the emergence of machine learning approaches. This position paper aims to update on these approaches.

State of the art works in OMR perform well with digitally written monophonic music, but there is plenty of room for improvement when it comes to reading handwritten music and complex pianoform scores [21, 22, 23]. The difficulty thus increases with the complexity of the music notation.

2. OMR PIPELINE

The standard OMR pipeline given by Rebelo et al. [37] is depicted in Figure 2:

1. Image preprocessing;
2. Music symbol recognition;
3. Musical information reconstruction;
4. Construction of a musical notation model.

In the first stage, images of sheet music are subject to techniques such as noise removal, binarisation, de-skewing and blurring in order to make the rest of the OMR processes more robust. Subsequently, reference lengths, such as staff lines thickness and distances between them are calculated. Typically, the next stage is musical symbol recognition. This stage consists of staff line processing and musical symbol processing and ends with classification. Primitives of musical symbols will be used in the third stage in order to reconstruct semantic meaning. Finally, all retrieved information should be embedded in an appropriate output file. A summary of these stages and the particular image processing and machine learning techniques employed in each stage are summarised in Table 2.

3. IMAGE PREPROCESSING

Image preprocessing is a fundamental step in many computer vision tasks. The primary outcome of this stage is an adjusted image that is easier to manipulate. Most common image manipulations include enhancement, de-skewing, blurring, noise removal and binarisation [2, 4, 39, 13, 29, 14, 26, 28, 22]. Image enhancement can include filters and adjusting the contrast or brightness for optimal object detection. De-skewing eliminates skewness and helps

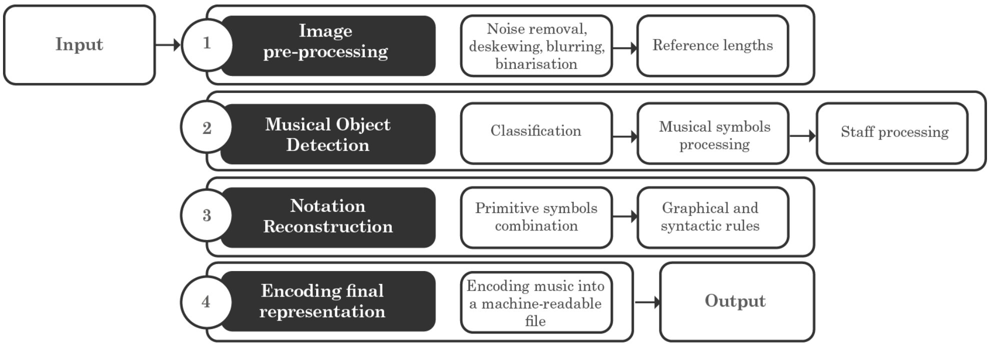

Input
1
Image pre-processing
Noise removal, deskewing, blurring, binarisation
Reference lengths
2
Musical Object Detection
Classification
Musical symbols processing
Staff processing
3
Notation Reconstruction
Primitive symbols combination
Graphical and syntactic rules
4
Encoding final representation
Encoding music into a machine-readable file
Output

Figure 2: Conventional OMR pipeline

Stage	Related Work
Image preprocessing	Fujinga [2, 38], Ng and Boyle [4], Fornés et al [39, 13], Tardón et al. [29], Pinto et al. [14], Calvo-Zaragoza et al. [26], Huang et al. [28], Wen et al. [22], Ridler et al. [40], Gocke [11], Ballard [41], Bainbridge and Bell [42], Cardoso et al. [43], Dalitz et al. [44]
Symbol Recognition	Mahoney [45], Prerau [34], Tardón et al. [29], Pacha [46], Rebelo et al. [12], Ng and Boyle [4], Choudhury et al. [47], Bainbridge and Bell [10], Fornés et al. [39, 13], Huang et al. [28] Fujinaga [2], Wen et al. [22], Pacha et al. [17, 24], Chen et al. [5], Gocke [11], Miyao and Nakano [48], Yun et al. [49]
Musical Information Reconstruction	Prerau [34], Pacha et al. [50, 51], Roy et al. [16], Bainbridge and Bell [10], Coüasnon et al. [3], Ng and Boyle [4], Baró et al. [52, 19] Calvo-Zaragoza et al. [21, 25, 27]
Musical Notation Model	Droettboom et al. [53], Chen et al. [5], Choudhury et al. [47], Ng et al. [9], Tardón [29], Bainbridge and Bell [10], Huang et al. [28]

Table 2: Summary of the studies carried in each of the OMR pipeline stages

in obtaining a more appropriate view in the object detection stage. Most of the digital images during the acquisition, transmission or processing are subject to noise. Both colour and brightness contain signals that carry random noise. Depending on the features of the image, different types of filters are used to remove some of the noise. During the process of binarisation, images are analysed to decide what is noise and what constitutes useful information for the task. Techniques to choose a binarisation threshold include global and adaptive methods. A global threshold is typically determined for the whole image, while for the adaptive threshold, local information in the image should be considered. Ng et al. [4] adapt the global threshold proposed by Ridler and Calvard [40]. While adaptive threshold is used in several more recent OMR studies too [13, 39, 11].

Gocke's [11] pipeline starts with a Gaussian filter, thenceforth a histogram on each colour channel is built. Then, the image is rotated to find the best angle that maximises horizontal projections. The image is then segmented into smaller 30x30 pixels, and a local threshold is found for each tile. Following threshold selection, all elements smaller than 4 pixels in diameter are removed, making the image clearer. The image is finally ready for staff-removal and symbol recognition. Local thresholding in this case yielded better results than the global one.

Similarly, in Fornés et al. [13] binarisation is followed by de-skewing using the Hough Transform [41]. A coarse approximation of the staff lines is obtained using median filters with horizontal masks to reconstruct the staff lines later. However, in this process, some residual colour information is retained, especially where the lines intersect with musical symbols, hence, some noise is still left. This approach is not robust to damaged paper.

Pinto et al. [14] propose a content-aware binarisation method for music scores. The model captures content-related information during the process from a greyscale image. It also extracts the staff line thickness and the vertical line distance in staff to guide binarisation. This algorithm tries to find a threshold that maximises the extracted content information from images. However, the performance hugely depends on the document characteristics, limiting performance across different documents.

Calvo-Zaragoza and Gallego [54, 55] propose using selectional auto-encoders [56] to learn an end-to-end transformation for binarisation. The network activation nodes indicate the likelihood of whether pixels are foreground or background pixels. Ensuing training, documents are parsed through the model and binarised using an appropriate global threshold. This approach performs better than the conventional binarisation methods in some document types. Nonetheless, errors happen around foreground strokes and are emphasised along edges of the input windows, due to the lack of context in the neighbourhood.

4. MUSIC SYMBOL RECOGNITION

The next stage typically constitutes dealing with musical symbol recognition. Here, the three main steps are staff processing, isolating music notations and finally, classification. Usually, staff lines are isolated first, then detected and finally removed from the images. The model then isolates the remaining notations as primitive elements. These are later used to extract features and feed those features to train the classifier.

4.1 Staff processing

Staff lines are a set of five horizontal paths from one side of the music score to the other. Each line and gap represent a different pitch. For better object detection, the question of staff line removal has been of prime importance. Researchers take two different approaches; one is only detecting and isolating them, while the other approach goes one step further in removing them.

While in printed sheet music, staff lines are straight, parallel and horizontal, in handwritten scores, these lines might be tilted, curved and may not be parallel at all. These lines might also look curved or skewed depending on the image skew angle [12] or the degradation of the paper. The model needs to separate staff lines from actual music objects. Since the lines overlap with musical objects, simply cutting and removing them degrades the notes and make them harder to recognise, further limiting performance.

Consequently, an increasing number of studies take the approach of removing the staff lines in a more intelligent fashion [4, 47, 57, 10, 39, 13, 29, 28, 22]. In this section, we outline typical staff line processing approaches. Blostein and Baird [57] suggests using horizontal projections of the black pixels and finding their maxima. The drawback is that the method only considers horizontal straight lines. In order to deal with non-horizontal, the process is followed with image rotations and choosing an angle with a higher maxima.

Rebelo et al. (2007) [12] consider staff lines to be the shortest path between two horizontal page margins if those paths have black pixels throughout the entire path. The height between every two lines is first estimated and later used as a reference length for the following operations. Upon choosing an estimation, using the Dijkstra algorithm [58], the shortest path between the leftmost pixel and the rightmost pixel is found. Their method is robust to lines with some curvature and discontinuity since it follows continuous paths connecting line ends from both sides. However, this algorithm may sometimes retain paths that do not follow the staff line. This happens when there is a higher density of beamed notes, and the estimated path follows the beams or when the staff lines are very curved.

Cardoso et al. [43] propose stable paths, considering the sheet music image as a graph. The staff lines in the graph are the less costly paths between the left and right margins. Subsequently, the model should differentiate between score pixels and staff line pixels. This model is robust to discontinuities, skewness, curvature in staff lines and one-pixel thin staff lines. Both the shortest path and stable paths give a similar false detection rate of 1.4% and 1.3% respectively. However, the stable path approach is five times faster. This technique is often used in the preprocessing stage [27].

Another study [7] uses stable paths approach to extract staff line skeletons. Then, the line adjacency graph (LAG) [59] is used to cluster pixel runs generated from run-length encoding (RLE) of the image [60]. The last step involves removing clusters lying on the staff line. This step has two passes; the first step estimates the height line for each staff by averaging the section height being cut with the staff lines. The second pass filters out the noise left from the last pass. This method takes a similar approach with [61] grouping staff line pixels into segments.

Other studies follow the approach of keeping the staff lines during the next stages [9, 11, 62, 26, 16, 63]. They argue that the staff line removal task is very complex and often ends up being inaccurate and passes errors to the following stages. These studies usually detect and isolate staff lines ahead of object processing. Recent object detection studies show that removing staff lines does not add much improvement to this stage [17].

A more recent work [46] investigates how incremental learning can assist staff line detection using convolutional neural networks (CNNs) and human annotation. To begin with, a CNN model is fed a small amount of data with available annotations for training. Using this training, the model makes predictions on a larger dataset, and a human annotator rejects or accepts the predictions. The accepted predictions are added to the training dataset to repeat the process. This method enables the creation of a more extensive dataset. After four iterations, the dataset contains 70% annotated scores of the original set. One drawback of incremental learning is that if the annotator accepts samples with imperfect annotations, the error accumulates in each iteration, introducing inaccuracy, while it also needs a human annotator. This yields similar results with [43, 61], with a precision score of 0.95.

Despite the substantial research effort put into staff line removal, it is still far from being accurate in handwritten sheet music. Handwritten scores exhibit a wide variety in line length and distance, thickness, curvatures of staff lines and also the quality of the image.

4.2 Music symbol processing

The next step after removing the staff lines is to isolate the musical symbols. Staff line removal will strongly affect this step as it can cause fragmentation in the parts where staff lines and musical objects are tangent to each other. One widely used approach is hierarchical decomposition [37], where staff lines split a music sheet and then extract noteheads, rests stems and other notation elements [47, 53, 11, 48, 4]. Some approaches consider, for instance, a half-note instead of its primitives for the classification step. Mahoney [45] uses descriptors to choose the matching candidate between a set of candidates of symbol types. Carter [36] uses the line-adjacency graph (LAG) of an image for both removing the staff lines and providing a structural analysis of symbols. This technique helps in ob-

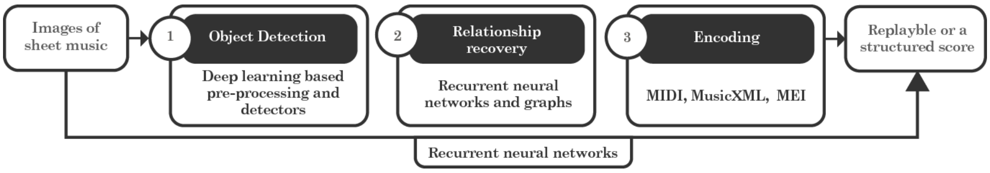

Images of sheet music
1
Object Detection
Deep learning based pre-processing and detectors
2
Relationship recovery
Recurrent neural networks and graphs
3
Encoding
MIDI, MusicXML, MEI
Replayble or a structured score
Recurrent neural networks

Figure 3: Typical OMR pipeline using deep neural networks

taining more consistent image sectioning, but it is limited to a small range of symbols as well as a potentially severe break-up of symbols.

Some studies skip segmentation and staff line removal [62, 63, 16] and use Hidden Markov Models (HMM). HMMs work on low-level features that are robust to poor quality images and can detect early topographic prints and handwritten pieces. Calvo-Zaragoza [63] split sheet music pages into staves following preprocessing. All staves are normalised and later represented as a sequence of feature vectors. This approach is very similar to [62], however, this study goes one step further and supports the HMM with a statistical N-gram model and achieve a 30% error rate. This performance could be further improved if lyrics are removed, light equalisation is performed and data variations are statistically modelled.

4.3 Music symbol classification

After the segmentation of musical primitives, the subsequent process is classification. Objects are classified based on their shapes and similarities. However, since these objects are very often densely packed and overlapping their shapes can become very complex. Therefore, this step is very sensitive to all possible variations in music notations. Fujinaga [2] uses projection profiles for classification, Gocke [11] uses template matching to classify the objects. Other methods used are support vector machines (SVMs), k-nearest neighbour (kNN), neural networks (NN) and hidden Markov models (HMM). A comparative study of the four methods [64], finds SVM performs better than HMMs.

Considering the success of deep neural networks (DNN) in many machine learning tasks, recent studies take this approach in music object recognition and classification. A typical pipeline is shown in Figure 3. These networks have many layers with activation functions employed before information propagates to the next layer. The deeper the model, the more complicated it gets and is able to detect hidden nonlinear relationships between the data, in this case, music objects. The problem with using DNNs in OMR is that they require a significant amount of labelled data for supervised training.

Object detection in images is a very active research field. Regional CNNs (R-CNNs), Faster R-CNN [65], U-nets [66], deep watershed detectors [18] and Single-shot detectors [67, 68] are among some of the approaches proposed recently. Pacha et al. [17] use Faster R-CNN networks with pre-trained models fine-tuned with data from Muscima++ (see Sect. 7 for a summary of OMR datasets). They achieve a mean average precision of up to 80 %. However, such performance is achieved with cropping the image into individual staff lines.

Tuggener et al. [18] use deep watershed detectors in the whole image. It is faster than Faster R-CNN approach in image snippets, and it allows some shift in the data distribution. Nonetheless, it does not perform well on underrepresented classes.

Going further into the pipeline, we should be able to capture and reconstruct the right positions, relationships between notes, and relevant musical semantic information such as duration, onsets, pitch.

5. NOTATION RECONSTRUCTION

After classifying and recognising musical objects, the next block should extract musical semantics and structure. As mentioned earlier, OMR is two-dimensional, meaning that recognising the note sequence as well as their spatial relationships are essential. Hence, a model should identify the information about the spatial relationship between the recognised objects. Ng et al. [9] believe that domain knowledge is key to improving OMR tasks and especially music object recognition, similarly to a trained copyist or engraver, to decipher poorly written scores, building on the authors' previous research on printed scores [4]. A multi-stage process is adopted, in which the first search is for essential features helping the interpretation of the score, verified by their mutual coherence, followed by a more intelligent search for more ambiguous features. Key and time signatures are detected after low-level processing and classification, using these global high-level features to test the earlier results.

Ng and Boyle [4] base their study on three assumptions: i) foreknowing the time signature, ii) key signature, and iii) that the set of the primitive feature set under examination is limited to ten. The first and second assumptions are overcome by geometrically predicting a limited symbol set such as numbers, flats and sharps. The input image goes through binarisation using a threshold, image rotation for de-skew, then the staff lines are detected and erased. Now the image has blocks of pixels, music object primitives and groups of primitives. Further segmentation based on some rules is needed for a group of primitives. After the segmentation process, a classifier uses only the width and the height of the bounding box for recognition based on a sampled training set. The recognised primitives are grouped to reconstruct their semantic meaning. The reconstruction consists of overlaying an ellipse and counting the number of foreground pixels, finding the pitch, search the neighbourhood for other features that might belong to the object and identifying the possible accidents using a nearest neighbourhood (NN) classifier. Music knowledge related to bars, time, and key signatures is applied at this stage. During segmentation, the process relies on straight edges of the objects, therefore is not robust to handwritten scores. The method fails if the symbols are skewed, for instance, when a stem is not perpendicular to a stave line.

Similar to the method mentioned above, another approach is formalising musical knowledge and/or encoding knowledge into grammar rules that explain, for instance, how primitives are to be processed or how graphical shapes are to be segmented [10, 3].

Prerau[34] proposes two levels of grammar. One being notational grammar while the other is a higher-level grammar for music. The first allows the recognition of symbol relationships, the second deals with larger music units. Many other techniques use musical rules to create grammar rules for OMR. Such rules can be exemplified as [37]:

- An accidental is placed before a notehead and at the same height;
- A dot is placed after or above a notehead in a variable distance;
- Between any pair of symbols: they cannot overlap.

The issue with music rules and heuristics is that these rules are very often violated, especially in handwritten music. Furthermore, it is challenging to create rules for many different variations and notations with a high level of complexity. As a result, this approach would not perform well with both typeset and handwritten complex notations, and it is difficult to scale to a broad range of notation and engraving styles.

Pacha et al. [50] propose using graphs to move towards a universal music representation. Considering that in music notations, the relationship between primitives contains the semantic meaning of each primitive; they suggest that OMR should employ a notation assembly stage to represent this relationship. Instead of using grammar and rules mentioned earlier, they use a machine learning approach to assemble a set of detected primitives. The assembly is similar to a graph containing syntactic relationships among primitives capturing the symbol configuration. The robustness of the model regarding variations in bounding boxes leaves room for improvement and so does the notation assembly stage, due to the lack of broader hypotheses on the detected objects.

Baró et al. [52] consider monophonic scores as sequences and use Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs) for reading such sequences to retrieve pitch and duration. For evaluation they use Symbol Error Rate (SER) defined as the minimum number of edit operation to convert an array to another. This approach shows to work well with simple scores such as monophonic scores, but fundamental remodelling is needed for more complex scores [52].

This stage reconstructs relationships, structure and semantics from the detected musical objects. A challenging problem in this stage is to model a musical output representation that encodes sheet music as a re-encoded score and the semantics (e.g. onsets, duration, pitch).

6. MUSIC NOTATION ENCODING

The output from the previous steps is used to construct a semantic model or data model. This model should represent a re-encoding of the score in the input. The output model should be expressible in a machine-readable format. Usual OMR output formats include MIDI, MusicXML, MEI, NIFF, Finale, and in some software, the music is even rendered into WAVE files. Musical Instrument Digital Interface (MIDI) [69] is an interchange medium between the computer and digital instruments. At the basic level, MIDI includes the temporal position when a note starts, stops, how loud the note is, the pitch of the note, instrument and channel. The main drawback of MIDI is that it cannot represent the relationships between musical symbols, or produce a re-encoded structured file, limiting the output to replayability only.

Notable formats that allow a structured encoding and storing notations include MusicXML [70, 71] and MEI [72, 73]. Both allow further editing in a music notation software. MusicXML is more focused on encoding notation layout. It is designed for archiving and for sharing sheet music between applications. There is ongoing research in the W3C Music Notation Community Group on improving MusicXML format to handle more specific tasks and applications.

The Music Encoding Initiative (MEI) [72] claims to be comprehensive, declarative, explicit and hierarchical. MEI has not been widely used as the final output of OMR systems yet. However, based on the characteristics mentioned above, MEI is able to capture and retain musical semantics better, e.g. relationships between voices, which may benefit music engraving.

There is also work converting OMR output into Semantic Web formats. Jones et. al. [74] propose the use of Linked Data to annotate and improve discovery of music scores using the Resource Description Framework (RDF). The captured information is limited to the number of voices, movements and melodies. Further extensions are needed to store more sophisticated music semantics that support harmony or melody analysis. Nevertheless, the use of Linked Data compatible formats may benefit OMR applications in multiple ways. Linking scores to other music related data on the Web [75] or even features of the audio of a performance [76] could support interactive applications such as score following or large catalogue navigation [77]. The ontologies governing these formats may be used to encode musical or engraving rules to complement probabilistic inference in machine learning models.

To decide which of the encodings to use, we have to think of what an application may require. Using the knowledge obtained in the previous steps and from different studies would assist this stage in its standardisation. Currently there is little research in OMR dealing with encoding, however, many works in other fields focus on encoding formats that better represent music and its structure.

7. DATASETS

Depending on the OMR task to be performed and the nature of the application, different datasets may be suitable. Existing datasets contain handwritten or copyright-free printed music sheets in mensural or CWMN notations. Calvo-Zaragoza et al. [78] introduced a new dataset called HOMUS (Handwritten Online Musical Symbols). This contains 15200 samples of 32 types of musical symbols from 100 different musicians. Universal Music Symbol Collection is a dataset of 90000 tiny handwritten and typeset music symbols from 79 classes that can be used to train classifiers.

As for staff line removal, a commonly used dataset is CVC-MUSCIMA [79]. It contains 1000 music sheets written by 50 different musicians. Each musician was asked to transcribe the same given 20 pages of music using the same pen and same style of sheet music paper. These pages include monophonic and polyphonic music, consisting of scores for solo instruments and music scores for choir and orchestra.

An upgraded version of CVC-MUSCIMA is MUSCIMA++ [15]. It is more suitable for musical symbol detection. It has 91255 symbols with both notation primitives and higher-level notation objects, key signatures or time signatures. Notes are captured using the annotated relationships of the primitives, having this way both low and high-level symbols. DeepScores is a collection that contains 300k annotated images of written music mainly for object classification, detection, and segmentation [80]. This dataset has large images containing tiny objects.

There are also datasets for an end-to-end recognition such as the Printed Images of Music Staves (PrIMuS) [21], or the extended version of this with distorted images to simulate imperfections Camera-PrIMuS [25]. These datasets have 87678 real-music scripts in five different formats: PNG, MIDI, MEI, semantic and agnostic encoding which is a sequence that contains the graphical symbols and their positions without any musical meaning.

Given that the performance of the deep learning methods usually depends on the amount of the data the model is fed, for future work, we propose creating a universal dataset that facilitates the intermediate stages but also an end-to-end system. We want to start by generating music files using a music notation software such as Dorico [81] or Rosegarden [82]. This work will be harmonized with the before-mentioned Muscima++ and DeepScores datasets.

8. OPEN ISSUES AND CONCLUSIONS

Low-quality images of sheet music, complex scores, handwritten music and alternate notations are still challenging for OMR, while most of the work focuses on monophonic scores. CWMN notation is highly complex, having dense scores, overlapping symbols, structural complexity, semantic rules that are sometimes violated. For a deep learning approach, in particular, class imbalance is one of the most significant issues; some note types are persistent while some others are rare. An further open issue is the lack of a large labelled dataset with a broad variety of image quality and balanced classes [83].

We can observe a shift in OMR from using conventional image processing and object detection to using neural networks, as shown in Figure 3. Recently published papers take novel approaches and use machine learning methods in all stages of the OMR pipeline. These stages are not necessarily in the order presented above or exhibit all the steps described.

Despite the introduction of deep learning, the field leaves space for improvement in all stages of the pipeline. New opportunities include creating more diverse and better balanced datasets, improving the detection of music objects and staff lines, the reconstruction of semantic meaning, and, perhaps most importantly, standardising the evaluation metrics and the output of the pipeline. A possible final goal is end-to-end learning that would not need intermediate steps. Neural networks are already applied to problems like text and speech recognition and machine translation in this manner. However, these systems are still not adapted to a two-dimensional output sequence such as music [31].

This paper summarised seminal and influential studies conducted in the field of OMR. We discussed different methods and approaches in prominent stages of the OMR pipeline. Our review aims to identify important older works and current state-of-the-art approaches, which can be used as a reference by researchers to begin further work in OMR. It also represents a position in several aspects of the field, including the need for incorporating more prior knowledge, theory and musical information in the processing pipeline, the need for finding new methods to incorporate these priors into statistical learning models such as deep neural networks and a need for more standardisation in OMR evaluation.

Acknowledgments

The authors acknowledge the support of the AI and Music CDT, funded by UKRI and EPSRC under grant agreement no. EP/S022694/1 and our industry partner Steinberg Media Technologies GmbH for their continuous support.

9. REFERENCES

[1] M. L. West, “The babylonian musical notation and the hurrian melodic texts,” *Music & letters*, vol. 75, no. 2, pp. 161–179, 1994.

[2] I. Fujinaga, “Optical music recognition using projections,” Ph.D. dissertation, McGill University Montreal, Canada, 1988.

[3] B. Coüasnon, P. Brisset, I. Stephan, and C. P. Brisset, “Using logic programming languages for optical music recognition,” in *In Proceedings of the Third International Conference on The Practical Application of Prolog*. Citeseer, 1995.

[4] K. C. Ng and R. D. Boyle, “Recognition and reconstruction of primitives in music scores,” *Image and Vision computing*, vol. 14, no. 1, pp. 39–46, 1996.

[5] G. Chen, L. Zhang, W. Zhang, and Q. Wang, “Detecting the staff-lines of musical score with hough transform and mathematical morphology,” in *2010 International Conference on Multimedia Technology*, Oct 2010, pp. 1–4.

[6] V. H. C. Vidal, "Staffline detection and removal in the grayscale domain," Ph.D. dissertation, Faculdade de Engenharia da Universidade do Porto, 2012.

[7] H.-N. Bui, I.-S. Na, and S.-H. Kim, "Staff line removal using line adjacency graph and staff line skeleton for camera-based printed music scores," in *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 2787–2789.

[8] Z. Huang, X. Jia, and Y. Guo, "State-of-the-art model for music object recognition with deep learning," *Applied Sciences*, vol. 9, no. 13, p. 2645, 2019.

[9] K. Ng, D. Cooper, E. Stefani, R. Boyle, and N. Bailey, "Embracing the composer: Optical recognition of handwritten manuscripts." in *ICMC*, 1999.

[10] D. Bainbridge and T. C. Bell, "A music notation construction engine for optical music recognition," *Softw., Pract. Exper.*, vol. 33, pp. 173–200, 2003.

[11] R. Gocke, "Building a system for writer identification on handwritten music scores," 2003. [Online]. Available: https://pdfs.semanticscholar.org/0bec/f150902ad48e080d0572dd5ef26686d442c6.pdf

[12] A. Rebelo, A. Capela, J. F. P. d. Costa, C. Guedes, E. Carrapatoso, and J. S. Cardoso, "A shortest path approach for staff line detection," in *Third International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution (AXMEDIS'07)*, Nov 2007, pp. 79–85.

[13] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke, "On the Use of Textural Features for Writer Identification in Old Handwritten Music Scores," in *2009 10th International Conference on Document Analysis and Recognition*. Barcelona, Spain: IEEE, 2009, pp. 996–1000. [Online]. Available: http://ieeexplore.ieee.org/document/5277541/

[14] T. Pinto, A. Rebelo, G. Giraldi, and J. S. Cardoso, "Music score binarization based on domain knowledge," in *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, 2011, pp. 700–708.

[15] J. Hajič and P. Pecina, "The muscima++ dataset for handwritten optical music recognition," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 39–46.

[16] P. P. Roy, A. K. Bhunia, and U. Pal, "Hmm-based writer identification in music score documents without staff-line removal," *Expert Systems with Applications*, vol. 89, pp. 222–240, 2017.

[17] A. Pacha, K.-Y. Choi, B. Couasnon, Y. Ricquebourg, R. Zanibbi, and H. Eidenberger, "Handwritten Music Object Detection: Open Issues and Baseline Results," in *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*. Vienna: IEEE, Apr. 2018, pp. 163–168. [Online]. Available: https://ieeexplore.ieee.org/document/8395189/

[18] L. Tuggener, I. Elezi, J. Schmidhuber, and T. Stadelmann, "Deep watershed detector for music object recognition," *arXiv preprint arXiv:1805.10548*, 2018.

[19] A. Baró, P. Riba, J. Calvo-Zaragoza, and A. Fornés, "From optical music recognition to handwritten music recognition: A baseline," *Pattern Recognition Letters*, vol. 123, pp. 1–8, 2019.

[20] A. Baro, P. Riba, and A. Fornés, "Towards the recognition of compound music notes in handwritten music scores," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE, 2016, pp. 465–470.

[21] J. Calvo-Zaragoza and D. Rizo, "End-to-end neural optical music recognition of monophonic scores," *Applied Sciences*, vol. 8, no. 4, p. 606, 2018.

[22] C. Wen, A. Rebelo, J. Zhang, and J. Cardoso, "A new optical music recognition system based on combined neural network," *Pattern Recognition Letters*, vol. 58, pp. 1–7, 2015.

[23] A. Pacha and H. Eidenberger, "Towards Self-Learning Optical Music Recognition," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Cancun, Mexico: IEEE, Dec. 2017, pp. 795–800. [Online]. Available: http://ieeexplore.ieee.org/document/8260732/

[24] ——, "Towards a universal music symbol classifier," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 2. IEEE, 2017, pp. 35–36.

[25] J. Calvo-Zaragoza and D. Rizo, "Camera-primus: Neural end-to-end optical music recognition on realistic monophonic scores." in *ISMIR*, 2018, pp. 248–255.

[26] J. Calvo-Zaragoza, I. Barbancho, L. J. Tardón, and A. M. Barbancho, "Avoiding staff removal stage in optical music recognition: application to scores written in white mensural notation," *Pattern Analysis and Applications*, vol. 18, no. 4, pp. 933–943, 2015.

[27] J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal, "Handwritten music recognition for mensural notation: formulation, data and baseline results," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 1081–1086.

[28] Y.-H. Huang, X. Chen, S. Beck, D. Burn, and L. J. Van Gool, "Automatic handwritten mensural notation interpreter: From manuscript to midi performance." in *ISMIR*, 2015, pp. 79–85.

[29] L. J. Tardón, S. Sammartino, I. Barbancho, V. Gómez, and A. Oliver, "Optical music recognition for scores written in white mensural notation," *EURASIP Journal on Image and Video Processing*, vol. 2009, no. 1, p. 843401, 2009.

[30] D. Byrd and J. G. Simonsen, “Towards a standard testbed for optical music recognition: Definitions, metrics, and page images,” *Journal of New Music Research*, vol. 44, no. 3, pp. 169–195, 2015.

[31] J. Calvo-Zaragoza, J. Hajič Jr., and A. Pacha, “Understanding Optical Music Recognition,” *arXiv:1908.03608 [cs, eess]*, Aug. 2019, arXiv: 1908.03608. [Online]. Available: http://arxiv.org/abs/1908.03608

[32] G. Jones, B. Ong, I. Bruno, and N. Kia, “Optical music imaging: music document digitisation, recognition, evaluation, and restoration,” in *Interactive multimedia music technologies*. IGI Global, 2008, pp. 50–79.

[33] D. Pruslin, “Automatic recognition of sheet music, 1966. a critical survey of music image analysis,” *Structured document image analysis. Springer, Heidelberg*, pp. 405–434, 1992.

[34] P. D, “Computer pattern recognition of standard engraved music notation,” in *Structured Document Image Analysis*. Springer, 1992, pp. 405–434.

[35] M. Kassler, “Optical character-recognition of printed music: A review of two dissertations,” 1972.

[36] N. P. Carter, “Automatic recognition of printed music in the context of electronic publishing.” Ph.D. dissertation, University of Surrey (United Kingdom)., 1989.

[37] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. S. Marcal, C. Guedes, and J. S. Cardoso, “Optical music recognition: state-of-the-art and open issues,” *Int J Multimed Info Retr*, vol. 1, no. 3, pp. 173–190, Oct. 2012. [Online]. Available: http://link.springer.com/10.1007/s13735-012-0004-6

[38] I. Fujinaga, “Staff detection and removal,” in *Visual Perception of Music Notation: On-Line and Off Line Recognition*. IGI Global, 2004, pp. 1–39.

[39] A. Fornés, J. Lladós, G. Sánchez, and H. Bunke, “Writer identification in old handwritten music scores,” in *2008 The Eighth IAPR International Workshop on Document Analysis Systems*. IEEE, 2008, pp. 347–353.

[40] T. Ridler, S. Calvard *et al.*, “Picture thresholding using an iterative selection method,” *IEEE transactions on Systems, Man and Cybernetics*, vol. 8, no. 8, pp. 630–632, 1978.

[41] D. H. Ballard, “Generalizing the hough transform to detect arbitrary shapes,” *Pattern recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[42] D. Bainbridge and T. Bell, “Identifying music documents in a collection of images,” in *ISMIR*. University of Canterbury. Computer Science and Software Engineering., 2006, pp. 47–52.

[43] J. dos Santos Cardoso, A. Capela, A. Rebelo, C. Guedes, and J. Pinto da Costa, “Staff Detection with Stable Paths,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1134–1139, Jun. 2009. [Online]. Available: http://ieeexplore.ieee.org/document/4775904/

[44] C. Dalitz, M. Droettboom, B. Pranzas, and I. Fujinaga, “A Comparative Study of Staff Removal Algorithms,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 753–766, May 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4359356/

[45] J. V. Mahoney, “Automatic analysis of music score images,” Ph.D. dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering ..., 1982.

[46] A. Pacha, “Incremental supervised staff detection,” in *Proceedings of the 2nd International Workshop on Reading Music Systems, Delft, 2019*. WORMs, 2019. [Online]. Available: https://alexanderpacha.files.wordpress.com/2019/11/2019-incremental-supervised-staff-detection.pdf

[47] G. S. Choudhury, M. Droetboom, T. DiLauro, I. Fujinaga, and B. Harrington, “Optical music recognition system within a large-scale digitization project.” in *ISMIR*, 2000.

[48] H. MIYAO and Y. NAKANO, “Note symbol extraction for printed piano scores using neural networks,” *IEICE TRANSACTIONS on Information and Systems*, vol. 79, no. 5, pp. 548–554, 1996.

[49] K. Yun, A. Huyen, and T. Lu, “Deep neural networks for pattern recognition,” *arXiv preprint arXiv:1809.09645*, 2018.

[50] A. Pacha, J. Calvo-Zaragoza, and J. Hajic jr, “Learning notation graph construction for full-pipeline optical music recognition,” in *20th International Society for Music Information Retrieval Conference*, 2019.

[51] A. Pacha and J. Calvo-Zaragoza, “Optical music recognition in mensural notation with region-based convolutional neural networks.” in *ISMIR*, 2018, pp. 240–247.

[52] A. Baró, P. Riba, J. Calvo-Zaragoza, and A. Fornés, “Optical music recognition by long short-term memory networks,” in *International Workshop on Graphics Recognition*. Springer, 2017, pp. 81–95.

[53] M. Droettboom, I. Fujinaga, and K. MacMillan, “Optical music interpretation,” in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 2002, pp. 378–387.

[54] J. Calvo-Zaragoza and A.-J. Gallego, “A selectional auto-encoder approach for document image binarization,” *Pattern Recognition*, vol. 86, pp. 37–47, 2018.

[55] A.-J. Gallego and J. Calvo-Zaragoza, “Staff-line removal with selectional auto-encoders,” *Expert Systems with Applications*, vol. 89, pp. 138–148, 2017.

[56] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, “Stacked convolutional auto-encoders for hierarchical feature extraction,” in *International conference on artificial neural networks*. Springer, 2011, pp. 52–59.

[57] D. Blostein and H. S. Baird, “A critical survey of music image analysis,” in *Structured Document Image Analysis*. Springer, 1992, pp. 405–434.

[58] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[59] S. Iliescu, R. Shinghal, and R. Y.-M. Teo, “Proposed heuristic procedures to preprocess character patterns using line adjacency graphs,” *Pattern recognition*, vol. 29, no. 6, pp. 951–975, 1996.

[60] T. Tsukiyama, Y. Kondo, K. Kakuse, S. Saba, S. Ozaki, and K. Itoh, “Method and system for data compression and restoration,” Apr. 29 1986, uS Patent 4,586,027.

[61] N. P. Carter and R. A. Bacon, “Automatic recognition of printed music,” in *Structured Document Image Analysis*. Springer, 1992, pp. 456–465.

[62] L. Pugin, “Optical music recognitoin of early typographic prints using hidden markov models.” in *ISMIR*, 2006, pp. 53–56.

[63] J. Calvo-Zaragoza, A. H. Toselli, and E. Vidal, “Early handwritten music recognition with hidden markov models,” in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE, 2016, pp. 319–324.

[64] A. Rebelo, G. Capela, and J. S. Cardoso, “Optical recognition of music symbols,” *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 13, no. 1, pp. 19–31, 2010.

[65] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[66] J. Hajic Jr, M. Dorfer, G. Widmer, and P. Pecina, “Towards full-pipeline handwritten omr with musical symbol detection by u-nets.” in *ISMIR*, 2018, pp. 225–232.

[67] J. Dai, Y. Li, K. He, and J. Sun, “R-fcn: Object detection via region-based fully convolutional networks,” in *Advances in neural information processing systems*, 2016, pp. 379–387.

[68] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[69] J. Rothstein, *MIDI: A comprehensive introduction*. AR Editions, Inc., 1995, vol. 7.

[70] M. Good, “Musicxml for notation and analysis,” *The virtual score: representation, retrieval, restoration*, vol. 12, pp. 113–124, 2001.

[71] M. Good and L. Recordare, “Lessons from the adoption of musicxml as an interchange standard,” in *Proceedings of XML*, 2006, pp. 5–7.

[72] P. Roland, “The music encoding initiative (mei),” in *Proceedings of the First International Conference on Musical Applications Using XML*, vol. 1060, 2002, pp. 55–59.

[73] A. Hankinson, P. Roland, and I. Fujinaga, “The music encoding initiative as a document-encoding framework.” in *ISMIR*, 2011, pp. 293–298.

[74] J. Jones, D. de Siqueira Braga, K. Tertuliano, and T. Kauppinen, “Musicowl: the music score ontology,” in *Proceedings of the International Conference on Web Intelligence*, 2017, pp. 1222–1229.

[75] G. Fazekas, Y. Raimond, K. Jakobson, and M. Sandler, “An overview of Semantic Web activities in the OMRAS2 Project,” *Journal of New Music Research special issue on Music Informatics and the OMRAS2 Project*, vol. 39, no. 4, pp. 295–311, 2011.

[76] A. Allik, G. Fazekas, and M. Sandler, “An ontology for audio features,” in *Proceedings of the International Society for Music Information Retrieval Conference*, 2016, pp. 73–79.

[77] L. Turchet, J. Pauwels, C. Fischione, and G. Fazekas, “Cloud-smart musical instrument interactions: Querying a large music collection with a smart guitar,” *ACM Transactions on the Internet of Things (In Press)*, 2020.

[78] J. Calvo-Zaragoza and J. Oncina, “Recognition of pen-based music notation: the homus dataset,” in *2014 22nd International Conference on Pattern Recognition*. IEEE, 2014, pp. 3038–3043.

[79] A. Fornés, A. Dutta, A. Gordo, and J. Lladós, “Cvc-muscima: a ground truth of handwritten music score images for writer identification and staff removal,” *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 15, no. 3, pp. 243–251, 2012.

[80] L. Tuggener, I. Elezi, J. Schmidhuber, M. Pelillo, and T. Stadelmann, “Deepscores-a dataset for segmentation, detection and classification of tiny objects,” in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 3704–3709.

[81] Steinberg, “Dorico,” https://new.steinberg.net/dorico/ (Jan. 2020).

[82] C. Cannam, R. Bown, and G. Laurent, “Rosegarden,” https://www.rosegardenmusic.com/ (Jan. 2020).

[83] J. Novotnỳ and J. Pokornỳ, “Introduction to optical music recognition: Overview and practical challenges.” in *DATESO*, 2015, pp. 65–76.