Hybrid Annotation Systems for Music Transcription

Ioannis Petros Samiotis*, Christoph Lofi†, and Alessandro Bozzon‡
Delft University of Technology
Email: *i.p.samiotis@tudelft.nl, †c.lofi@tudelft.nl, ‡a.bozzon@tudelft.nl

***Abstract*—Automated methods and human annotation are being extensively utilized to scale up modern classification systems. Processes though such as music transcription, oppose certain challenges due to the complexity of the domain and the expertise needed to read and process music scores. In this work, we examine how music transcription could benefit from systems that utilize hybrid annotation workflows, where automated methods are being trained, evaluated or have their output fixed by crowdworkers, using microtask designs. We argue that through careful task design utilizing microtask crowdsourcing principles, the general crowd can meaningfully contribute to such hybrid transcription systems.**

***Index Terms*—crowd computing, crowdsourcing, music transcription, hybrid annotation systems**

I. Introduction

Traditionally, digital music transcription involves highly trained experts who need to understand music structures and notations. Among their skills are also the use of specialised software tools and the ability to identify and fix errors of previous editions. Through Optical Music Recognition (OMR), researchers try to automate the process (or parts of it), through several processing steps such as score segmentation, symbol recognition and semantic reconstruction of a scanned music score.

State-of-the-art methods show acceptable performance in the case of clean music scores, but their quality quickly degrades in case of hand written notes [1]. In general, they still require substantial human intervention to provide results with consistent quality [1], [2], while interactive systems that could utilize human evaluation in an efficient and scalable way are still an open issue [3].

Microtask crowdsourcing is a popular approach for scaling up digital content annotation tasks. On online microtask crowdsourcing platforms, such as Amazon Mechanical Turk, large groups of individuals - called workers - perform *microtasks* like image categorization, and audio or text transcription. By splitting a complex and cognitively intensive task into simpler steps, *microtasks* crowdsourcing allows people with little to no expertise, to contribute to knowledge-intensive activities [4].

Explicit control over a crowd's product, is in the heart of microtask crowdsourcing [5], [6]. To that end, microtasks design should allow the measurement of their outcomes in an algorithmic fashion. Few studies addressed the use of microtask crowdsourcing for music scores transcription, and they typically focus on guiding the workers in the transcription of whole scores [7] or by providing support to the experts [8], [9]. However, music scores are complex artefacts that need specific domain knowledge to read and understand, making the task of transcribing a score complex and cognitively demanding. To the best of our knowledge, how to address the task of score transcription through microtask crowdsourcing remains an open research question [10].

In this paper we build upon our preliminary work [11] which shows the general feasibility of microtask crowdsourcing for error detection in music transcription. We showcase how hybrid annotation systems that utilize both OMR processes and microtask crowdsourcing could be designed and we discuss their feasibility.

II. Related work

The topic of microtask crowdsourcing for music transcription is scarcely addressed in literature, with many relevant research questions left unanswered. In Burghardt et al. [7] the *Allegro* system was developed, a tool to allow the transcription of entire scores by a (single) human worker. However, *Allegro* has only been tested on a limited number of users, and it was not deployed on an online microtask crowdsourcing platform. The same limitation holds for the work in [8], one of the first attempts to study human input and how the task design can affect human input. This study focused on analysing segments which are one measure long, which is the smallest unit of analysis in our study as well. We expand this, by studying also how the size of the segment shown to the crowd affect its performance. An important work to mention is OpenScore [12], up to now the largest scale project to incorporate humans in music score transcription. In terms of user participation though, it was mainly carried out by seven community members with extensive musical background. Moreover they report different issues related to the management of data (done manually by the administrators of the platform) and user engagement (without any control they would focus on their preferred music score) admitting in the end that in their project "OMR (involving humans) is not currently a scalable solution".

So far, there is not any literature that has targeted unknown crowds with varying skills for music transcription tasks, thus research questions on [10] about what type of tasks users can perform and how to evaluate them still remain open. In this work we address this research gap by looking into similar crowdsourcing works in other domains. More specifically, in [13] it was found that for knowledge-intensive tasks involving artworks, a crowd with varying and unknown domain-specific knowledge found on online platforms can produce useful annotations when aided by good task design.

Research has shown that UI design is an important part of a microtask design [14]. Research so far has experimented with various designs such as showing spectogram visualisations for audio annotation [15] or the use of chat-bots to assist common types of microtasks [16], all of which have yielded positive results on the performance of the crowdworkers.

III. Hybrid Music Transcription Workflows

In a hybrid annotation workflow, the goal is to effectively and efficiently combine automated methods with people, to achieve a result that couldn't be attained otherwise, by the separate methods themselves, as we can see in [17]. This means that the individual steps of the workflow need to be identified early on and through careful design, to allocate each step to the appropriate processing method. Automated methods and human input need to co-exist in such systems and complement each other and for a hybrid music transcription workflow, we will need to factor the complexity and niche knowledge required by a person to transcribe a music score but also potential shortcomings of OMR methods. More specifically, hybrid annotation workflows cover three main components: Algorithms / Machine Learning, the Crowd, and a Quality Assessment mechanism.

A. Hybrid Workflow Patterns

Algorithms / Machine Learning: A set of algorithms, typically machine learning algorithms, which can process the input data into the desired output data. However, these algorithms typically have at least one of the following two shortcomings: The results produced by the algorithms are of bad quality and insufficient for the desired use. The algorithm relies on extensive training which is typically not or only partially available.

The first problem can manifest in different ways. For instance, the algorithm could be good enough in most cases, but might fail in certain cases. Here, it would be necessary to identify when the algorithm failed (automatically or using crowdsourcing) and provide the failed data by using crowdsourcing. The extreme worst case of this would be a scenario in which the algorithm fails in all cases. In case of a failure, the crowd needs to redo the algorithms output. Alternatively, the algorithms generally works on any kind of input data, but the output quality is slightly too low in nearly all cases. Here, all outputs need to be slightly adjusted and fixed using crowdsourcing. In case of a failure, the crowd needs to improve the algorithms output. The second problem requires the creation of training data that usually covers a large number of examples of correct input data / desired output data pairs. Crowdworkers can provide such training pairs upfront for initial training, or as part of the fixing measures introduced for the first problem. Then, this crowd-provided data can be used for incremental re-training.

The Crowd: The crowd can be used to execute cognitive Human Intelligence Tasks. The actual choice of crowdworkers and their incentivisation is a core challenge of other deliverables, but this could range from paying microtask workers on platforms like Amazon Mechanical Turk to motivating expert online communities using intrinsic incentives. In general, the crowd can be used to: Check the correctness of an intermediate algorithm result. This can range from simple correct / incorrect checks to more complex checks which give a detailed overview of the location and nature of the error. Produce results: Here, the crowd is used to perform the same task the algorithm was designed for: transform a given input data instance into the correct output data. This functionality is employed when an algorithm failed to process, or when that data is required for further/initial training. Improve results: Here, a machine-produced result with sub-par quality is manually improved. Typically, this should be employed when improving slightly faulty outputs is easier and cheaper than creating a new output manually from scratch.

Quality Assessment: This is a core component to ensure the effectiveness of a good hybrid crowdsourcing process. The quality assessment is central in both judging the quality of algorithmic results in order to decide if and what kind of crowd treatment is needed, but also for judging the reliability and quality of crowd feedback in light of low-skill and/or malicious workers.

B. OMR Processes and Challenges

To better understand how hybrid annotation patterns could transfer to music transcription, we need to first identify OMR processes that are being used. Processing steps for OMR have been identified in [18] where we group processes in the following three main categories:

- Image pre-processing
- Music symbol segmentation and recognition
- Semantic Reconstruction

During the image pre-processing part, different techniques are often applied to scanned images for reducing the computational cost and making the next OMR steps more efficient. One of the most important methods of image preprocessing is "Binarization" which is the process of converting the pixel image into a binary image (black and white), separating the foreground from the background. This is a common step for most of the OMR tools, and the next steps depend on it. Binarization eases the OMR tasks and reduces the amount of information the following steps need to process. For example, it is easier to detect a music symbol in a binary image than in a color image. However, binarization can also pollute the image [18]. Furthermore, many music scores are ancient documents in poor condition due to paper degradation (yellowing, mold, and mildew, etc.), and this often introduces noise on the image, reducing the quality of the OMR tool output. Therefore, working with old music score sheet requires a specialized algorithm for image-clean and binarization to reduce the aforementioned problems [19].

Music symbol segmentation is the process of locating and isolating the music object. The main objective is to find the correct position of each symbol to be identified in the next OMR step. This is one of the most challenging OMR steps and highly error-prone. Most of the symbols on a music score

are connected by staff lines. In order to isolate those symbols, staff lines must be detected, and then removed. An accurate staff line removal is a challenge because symbols have to be disconnected from the staff line without removing pixels belonging to the symbols themselves. Unfortunately, staff lines are not always perfectly horizontal, so it requires knowing the exact location of the staff line at each horizontal coordinate. This procedure can be even more complex due to poor music score images (paper degradation, stains, etc), and zones with a high density of symbols [20]. After the segmentation stage, the segmented symbols need to be recognized and classified into symbols of predefined groups, such as notes, rests, accidentals, clefs, etc. Symbol identification is a hard task due to symbol variability. Each symbol can have different variations considering different score editors or the continuous evolution of music notation over time. However, variability can be also observed in the same music score, making it even more difficult to model each symbol due to ambiguity. Besides that, the previous segmentation step may cut the objects, making the recognition even more difficult [20].

The last stage of the OMR framework is to reconstruct the music semantics from the previous recognized symbols. Basically, the recognized graphical symbols must be combined with the staff system to reproduce the meaning of the scanned music [18]. Unlike optical character recognition (OCR) which is predominantly one-dimensional, OMR tools require an interpretation of two-dimensional relationships between music objects. As a consequence many errors may occur due to a symbol placed in a wrong position. For example, a slur symbol is a curved line generally located over the notes indicating that the notes it embraces have to be played without separation. If a slur is placed in a wrong position, it leads to a misinterpretation of the music score. Likewise, a dot has different meaning depending on where it is located. The ultimate step of OMR systems is to export the final semantic score into a final representation of musical score. Several formats have been developed, such as MIDI (Musical Instrument Digital Interface), MusicXML, MEI (Music Encoding Initiative), NIFF (Notation Information File Format), etc. Generally, each tool has its own output format or a set of output formats. This lack of a commonly accepted representation imposes an obstacle for OMR tool assessment [21].

IV. Case Study: Error Detection

As a case study, we'll showcase the experiment conducted in [11]. Through that preliminary work, we research how microtask crowdsourcing design can be implemented on music transcription. Such workflow could fit, in a hypothetical hybrid annotation system, during the training or evaluation step of an OMR algorithm. The main focus of that work was to study to what extent a general crowd can identify *errors* in a music score transcription. The experiment aimed at testing the ability of crowd workers to spot errors using interfaces having a combination of visual and audio components.

A. Task Design

This study aimed on how different task design factors can influence the crowdworkers performance, focusing on two aspects:

1) The *modality* (*visual* versus *audio*) used to spot errors: as music scores are complex artefacts, and music is primarily an auditory experience. Therefore, it was investigated how the score comparison *modality* affect the error detection performance in workers that are potentially not familiar with musical notation. Intuitively, the interest was if "hearing" errors is easier that "seeing" errors.
2) The score *size* offered to crowdworkers for annotation. The goal was to assess how the size (in terms of measures) of the score offered to worker affects their performance.

B. Dataset Creation

A single classical music score was used to avoid introducing additional variable in workers' performance. Specifically the study used the Urtext of "*32 Variations in C minor*" by Ludwig van Beethoven. It is a piano piece and the music artifacts are all printed typeset forms. This is a slightly easier use case than hand-written scores. The score was retrieved from IMSLP as a PDF[1].

As a Gold standard transcription of that PDF we used an MEI[2] file that had been transcribed by an expert. This file was accepted as error free, and it allowed errors to be introduced in a controlled way for the experiments.

The music score was segmented in varying sizes to investigate how workers cope with shorter or longer tasks. We distinguish 1) *one measure* segments, 2) segments of *two measures* and 3) segments of *three measures*. Both of the two digital versions of the score, the PDF file of the original score and the transcribed MEI file, were segmented using the aforementioned segment sizes.

The errors that were introduced to the MEI segments, derived from common errors that can occur in automatic OMR systems. The type of errors could impact the crowdworkers' ability to spot them and correctly identify them as errors. Therefore, different types of errors were studied, all focusing on the music notes themselves and their accidentals. Errors on performance annotations, clefs, finger numbers etc, were out of scope in that study. The following types of error were introduced per MEI segment: 1) *Missing notes*; 2) *Wrong vertical position of a note*; 3) *Wrong duration of a note*; 4) *Wrong accidental*.

C. User Interface Design

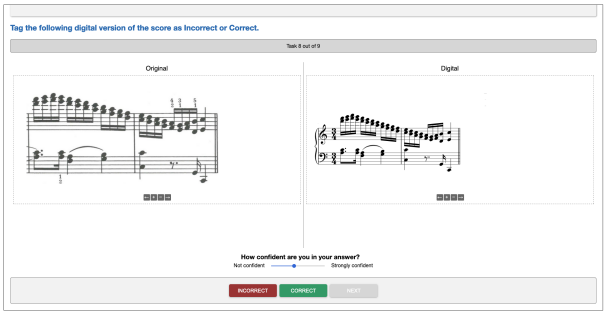
These design considerations resulted in the following three interface designs. Each combination of interface with a segment size consists of a microtask:

noitemsep

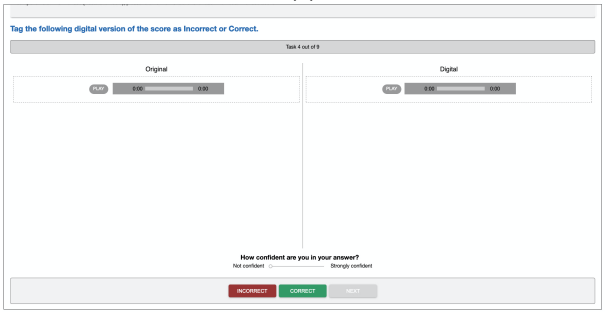
[1]https://imslp.org/wiki/32_Variations_in_C_minor%2C_WoO_80_(Beethoven%2C_Ludwig_van)

[2]https://music-encoding.org/

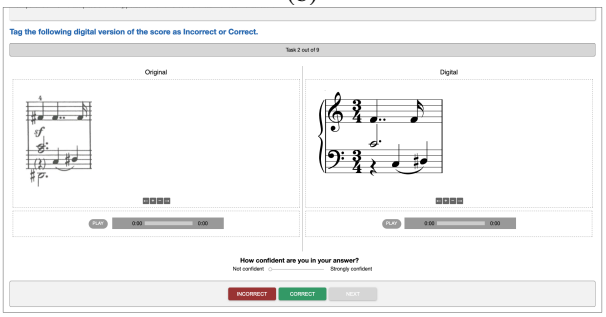
- **Original Score** against **Correct/Incorrect MEI Render (Visual)**: This user interface, depicted in Figure 1(a), shows the segment of the original scanned score to the left, with the corresponding MEI render to the right. The user needs to compare the two images and spot differences related to the types of errors.
- **Correct MIDI** against **Correct/Incorrect MIDI (Audio)**: In this interface, as shown in Figure 1(b), we let the user listen to the correct MIDI extract on the left and the one generated from the MEI transcription to the right.
- **Original Score and Correct MIDI** against **Correct/Incorrect MEI and Correct/Incorrect MIDI (Combination)**: This final user interface, as shown in Figure 1(c), combines elements of the previous two. The user here has the option to either use the visual comparison, the audio comparison or both to realise if there are errors to the segment to the right. The MEI render and MIDI extraction to the right always originate from the same MEI transcription, therefore both will be correct or both will contain errors.



(a)



(b)



(c)

Fig. 1. Microtask User Interfaces: (a) Visual, (b) Audio and (c) Combination

D. Results

In total, 144 workers executed our tasks on MTurk and we paid them per task execution according to the average US minimal hourly wage[3]. In order to minimize the effect of any biases or learning effect we randomized the order of the presentation of the different task designs (UI-segment size combination). One worker eluded the quality verification on task interface, which results in 143 unique workers.

As expected, people with some formal knowledge in music, which could be useful to comprehend music scores, are very rare "in the wild". To enable the use of microtask crowdsourcing for music score transcription, good task design is therefore of essence. We refer to [11] for more detailed analysis, but overall the results show that error detection is a task that could be successfully performed in a microtask crowdsourcing setting. Offering audio extracts of a target music score can positively affect the performance of the crowdworkers, especially for short segments of one or two measures. With larger segments, even though audio extracts are still yielding better results against to the textual measures of the score, a combination of the two modalities is more preferable. This result gives important indications for task splitting and scheduling purposes, as it suggests that it is possible to evaluate larger portions of scores without incurring accuracy penalties. This has obvious implications in terms of overall transcription costs.

V. Conclusion

Crowdsourcing and human computation are powerful tools which can be integrated in a data processing pipeline or information system to handle processing tasks which cannot easily be covered by current algorithmic approaches due to the involved semantic complexity. However, crowdsourcing is expensive: workers need to be incentivised (often with monetary incentives, or carefully engineered social incentives), and human work is of course often slower than automated algorithms. This gives a strong argument to strive for hybrid crowdsourcing workflows, where algorithms and humans work hand in hand. Such systems get the best of both worlds: the efficiency of algorithms and the cognitive power and insight of humans.

References

[1] B. Almeida and S. Spanner, "Allegro: User-centered Design of a Tool for the Crowdsourced Transcription of Handwritten Music Scores," in *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage*, vol. 25, no. 23. New York, New York, USA: ACM Press, 2017, pp. 15–20. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3078081.3078101

[2] P. Bellini, I. Bruno, and P. Nesi, "Assessing Optical Music Recognition Tools," *Computer Music Journal*, vol. 31, no. 1, pp. 68–93, mar 2007. [Online]. Available: http://www.mitpressjournals.org/doi/10.1162/comj.2007.31.1.68

[3] J. Calvo-Zaragoza, J. H. Jr., and A. Pacha, "Understanding optical music recognition," *ACM Comput. Surv.*, vol. 53, no. 4, Jul. 2020. [Online]. Available: https://doi.org/10.1145/3397499

[3]We estimated an average task completion time of 15'; each crowdworker was awarded 2.5$

[4] J. Oosterman, J. Yang, A. Bozzon, L. Aroyo, and G.-J. Houben, "On the impact of knowledge extraction and aggregation on crowdsourced annotation of visual artworks," *Computer Networks*, vol. 90, pp. 133 – 149, 2015, crowdsourcing. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128615002315

[5] E. Law and L. v. Ahn, "Human computation," *Synthesis lectures on artificial intelligence and machine learning*, vol. 5, no. 3, pp. 1–121, 2011.

[6] A. Bozzon, M. Brambilla, S. Ceri, and A. Mauri, "Reactive crowdsourcing," in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 153–164.

[7] M. Burghardt and S. Spanner, "Allegro: User-centered design of a tool for the crowdsourced transcription of handwritten music scores," in *Proceedings of the 2Nd International Conference on Digital Access to Textual Cultural Heritage*, ser. DATeCH2017. New York, NY, USA: ACM, 2017, pp. 15–20. [Online]. Available: http://doi.acm.org/10.1145/3078081.3078101

[8] L. Chen and C. Raphael, "Human-Directed Optical Music Recognition," *Electronic Imaging*, vol. 2016, no. 17, pp. 1–9, feb 2017. [Online]. Available: http://www.ingentaconnect.com/content/10.2352/ISSN.2470-1173.2016.17.DRR-053

[9] L. Chen, R. Jin, and C. Raphael, "Human-Guided Recognition of Music Score Images," in *Proceedings of the 4th International Workshop on Digital Libraries for Musicology - DLfM '17*. New York, New York, USA: ACM Press, 2017, pp. 9–12. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3144749.3144752

[10] C. Saitis, A. Hankinson, and I. Fujinaga, "Correcting large-scale omr data with crowdsourcing," in *Proceedings of the 1st International Workshop on Digital Libraries for Musicology*, 2014, pp. 1–3.

[11] I. Samiotis, S. Qiu, A. Mauri, C. Liem, C. Lofi, and A. Bozzon, "Microtask crowdsourcing for music score transcriptions: an experiment with error detection," in *Proceedings of the 21st International Society for Music Information Retrieval Conference*, 2020.

[12] M. Gotham, P. Jonas, B. Bower, W. Bosworth, D. Rootham, and L. VanHandel, "Scores of scores: an openscore project to encode and share sheet music," in *Proceedings of the 5th International Conference on Digital Libraries for Musicology*, 2018, pp. 87–95.

[13] J. Oosterman, A. Bozzon, G.-J. Houben, A. Nottamkandath, C. Dijkshoorn, L. Aroyo, M. H. Leyssen, and M. C. Traub, "Crowd vs. experts: nichesourcing for knowledge intensive tasks in cultural heritage," in *Proceedings of the 23rd International Conference on World Wide Web*, 2014, pp. 567–568.

[14] U. Gadiraju, A. Checco, N. Gupta, and G. Demartini, "Modus operandi of crowd workers: The invisible role of microtask work environments," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–29, 2017.

[15] M. Cartwright, A. Seals, J. Salamon, A. Williams, S. Mikloska, D. MacConnell, E. Law, J. P. Bello, and O. Nov, "Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations," *Proceedings of the ACM on Human-Computer Interaction*, vol. 1, no. CSCW, pp. 1–21, 2017.

[16] P. Mavridis, O. Huang, S. Qiu, U. Gadiraju, and A. Bozzon, "Chatterbox: Conversational interfaces for microtask crowdsourcing," in *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, 2019, pp. 243–251.

[17] C. Lofi and K. El Maarry, "Design patterns for hybrid algorithmic-crowdsourcing workflows," in *2014 IEEE 16th Conference on Business Informatics*, vol. 1. IEEE, 2014, pp. 1–8.

[18] A. Rebelo, I. Fujinaga, F. Paszkiewicz, A. R. Marcal, C. Guedes, and J. S. Cardoso, "Optical music recognition: state-of-the-art and open issues," *International Journal of Multimedia Information Retrieval*, vol. 1, no. 3, pp. 173–190, 2012.

[19] B. Gatos, I. Pratikakis, and S. J. Perantonis, "An adaptive binarization technique for low quality historical documents," in *International Workshop on Document Analysis Systems*. Springer, 2004, pp. 102–113.

[20] F. Rossant and I. Bloch, "Robust and adaptive omr system including fuzzy modeling, fusion of musical rules, and possible error detection," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–25, 2006.

[21] 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.