PatternFinder: Content-based music retrieval with music21

David Garfinkle
McGill University
Monday, August 7 2017

Symbolic Content-Based Music Retrieval (CBMR)

Goal: Given a query (pattern) of symbolic music, find all of the similar occurrences of this query within a database (source)

Why: Computer-aided musicology of symbolic music scores

Challenges:

- Application-dependent task
- Polyphonic music searching

PatternFinder

- Python package built on top of music21
- We started by implementing seven CBMR algorithms developed at the University of Helsinki (Kjell Lemström, Antti Laaksonen, Esko Ukkonen, Mika Laitinen)
- These algorithms find music similarity by trying to intersect sets of two-dimensional points

Piano-roll Example

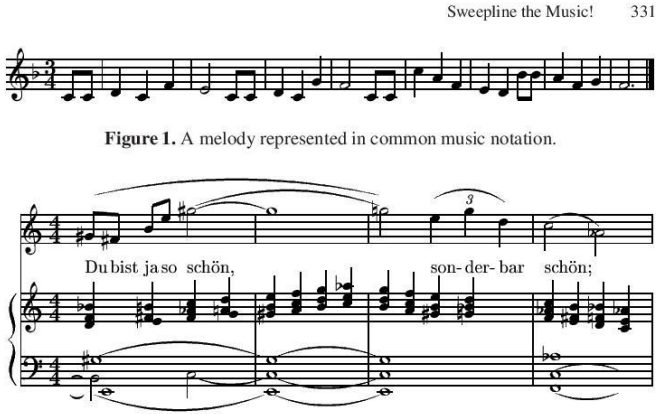


Figure 1. A melody represented in common music notation.

Figure 2. An excerpt of Einojuhani Rautavaara's opera *Thomas* (1985). Printed with the permission of the publisher Warner/Chappell Music Finland Oy.

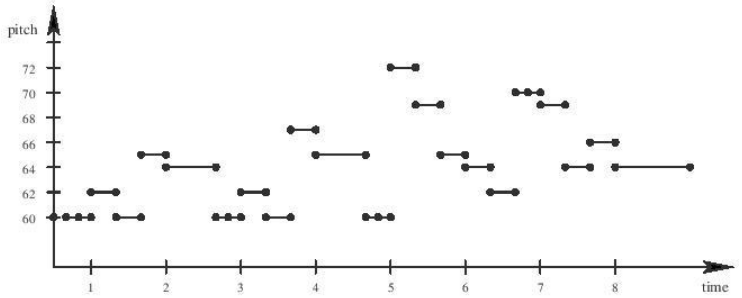
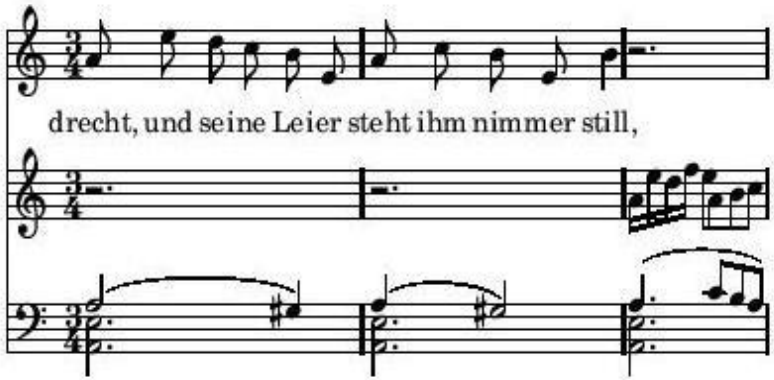


Figure 3. The example of Fig. 1 in piano-roll representation.

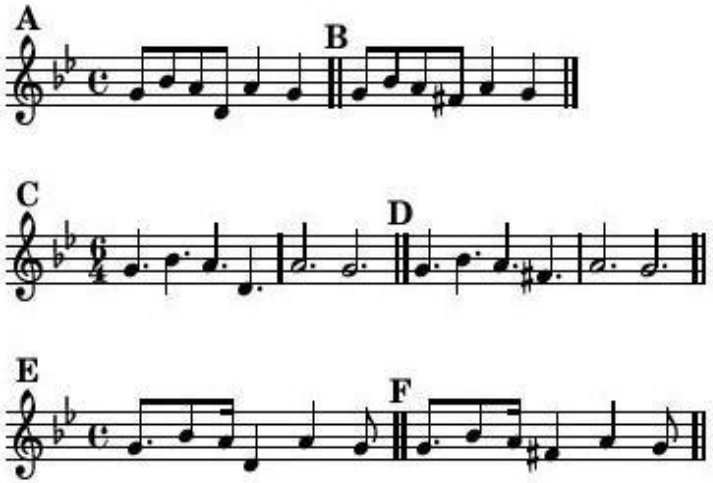
Ukkonen, E., Lemström, K., & Mäkinen, V. (2003). Sweepline the Music! *Lecture Notes in Computer Science*, *2598*, 330–342.

Source



Excerpt from Schubert's *Der Leiermann*

Queries

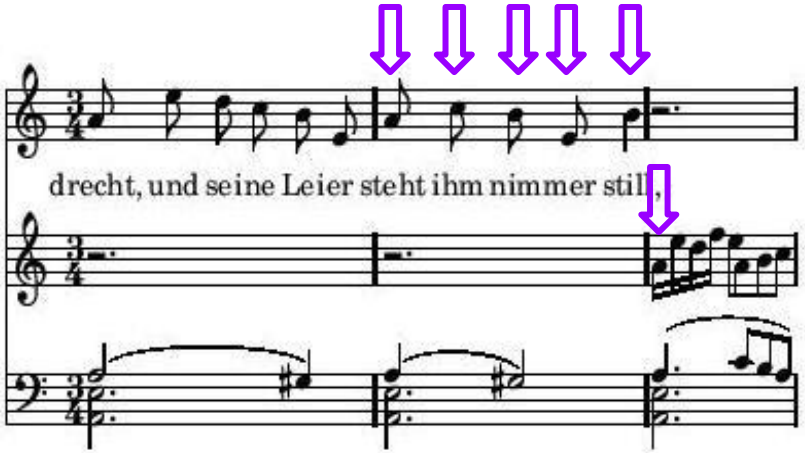


K. Lemstrom and M. Laitinen. Transposition and time-warp invariant geometric music retrieval algorithms. In Proc. ADMIRE'11, Third International Workshop on Advances in Music Information Research, Barcelona, 201

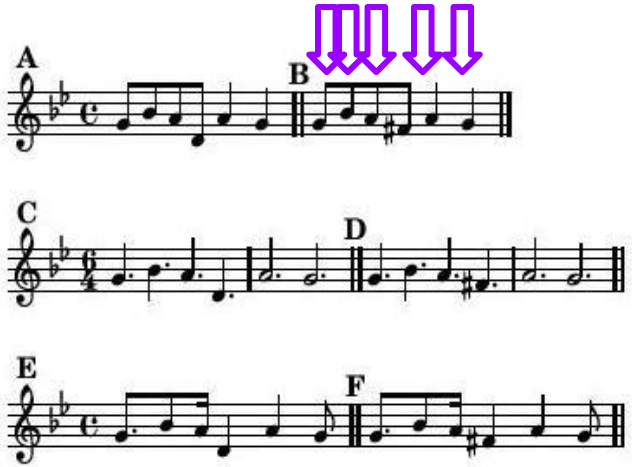
Threshold

- Minimum number of notes in the pattern which get mapped somewhere in the database
- Exact (every note is matched) or approximate (at least x pattern notes are matched)
- Or one can specify ‘mismatches’, meaning at most x pattern notes are missed

Threshold



Excerpt from Schubert's *Der Leiermann*



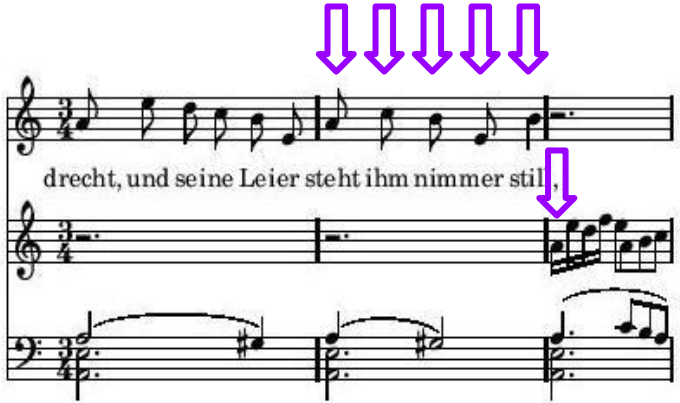
Queries B, D, and F require a threshold of at least 5 (or 85%)

Queries A, C, and E require a threshold of at least 6 (or 100%)

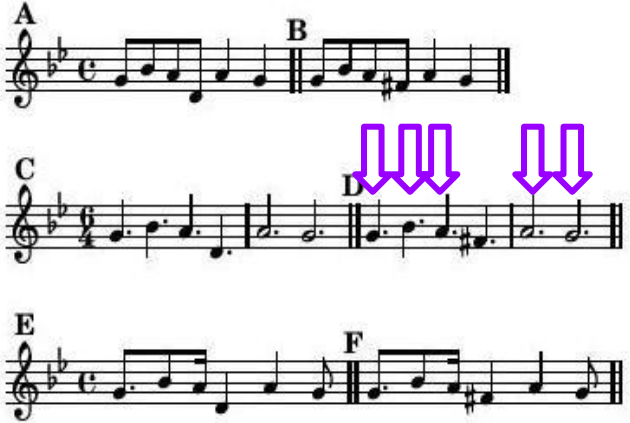
Scale

- Time-scaling liberties taken by the algorithm to find a match
- *Pure*: rhythmically identical occurrences
- *Scaled*: finds augmentation and diminution
- *Warped*: rhythmic values are ignored

Scale



Excerpt from Schubert's *Der Leiermann*



Queries A and B require a scale of 1

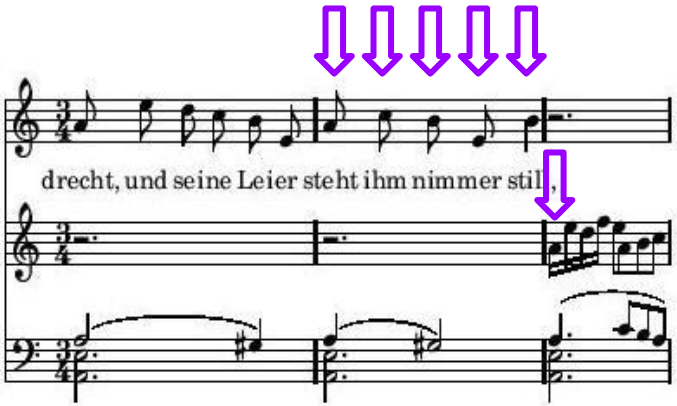
Queries C and D require a scale of 3/2

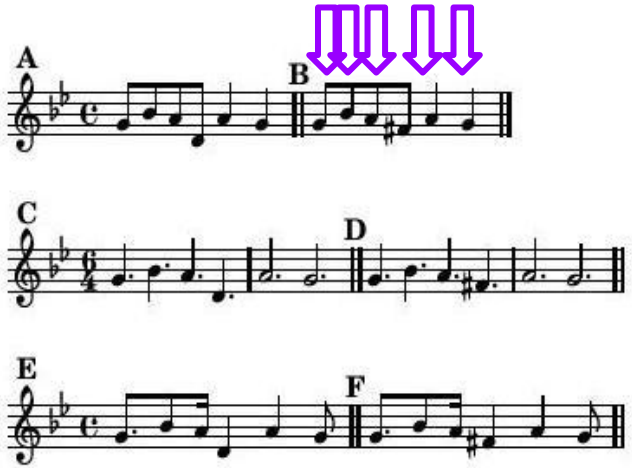
Queries E and F require 'warped'

Window

- Number of intervening notes allowed between two matched notes
- Pattern window
- Source window

Window





All queries would require a source window of 4

Queries B, D, F need a pattern window of at least 2

Limitations and Future Work

- Ranking system
- Implementation of popular monophonic search methods (which are comparatively more effective than polyphonic-capable algorithms in their domains)
- Implement index and filtering methods for scalable database queries

SIMSSA | Single Interface for Music Score Searching and Analysis

Social Sciences and Humanities Research Council of Canada

Conseil de recherches en sciences humaines du Canada

Canada

McGill

Schulich School of Music
École de musique Schulich

DDMAL DISTRIBUTED DIGITAL MUSIC ARCHIVES & LIBRARIES LAB

CIRMMT Centre for Interdisciplinary Research in Music Media and Technology

Fonds de recherche Société et culture
Québec

References

Ukkonen, E., Lemström, K., & Mäkinen, V. (2003). Sweepline the Music! *Lecture Notes in Computer Science*, *2598*, 330–342.

K. Lemstrom and M. Laitinen. Transposition and time-warp invariant geometric music retrieval algorithms. In Proc. ADMIRE'11, Third International Workshop on Advances in Music Information Research, Barcelona, 201

Lemström, K. (2010). Towards More Robust Geometric Content-Based Music Retrieval. In J. S. Downie & R. C. Veltkamp (Eds.), *Proceedings of the 11th International Society for Music Information Retrieval Conference* (pp. 577–582). Utrecht, Netherlands.

Laaksonen, A. (2013). Efficient and Simple Algorithms for Time-Scaled and Time-Warped Music Search. In *Proceedings of the 10th International Symposium on Computer Music Multidisciplinary Research* (pp. 621–630). Marseille, France: Laboratoire de Mécanique et d'Acoustique.