C
F
G
C
F
G7
C
C
F
G
C
F
G7
C
C
F
G
G7
C
Am
Dm
G7
C
F
G
1 C
2 C

Optical Music Recognition

Elin Lager, elila660
Josefine Flügge josfl888

December 16, 2018

Abstract

This project was made in the course Advanced Image Processing (TNM034) at Linköping University. The aim of the project is to create a program for Optical Music Recognition. The program is written in Matlab and takes scanned images of sheet music as input and returns a string representing the detected quarter and eight notes.

Contents

1 Introduction

1.1 Purpose and Aim

The purpose of this project is to create a program that can take an image of sheet music as input and give a string representing the notes as output.

1.2 Delimitations

The system is delimitated to only detect quarter notes and eigth notes and ignore all other notes, rests, dots, ties and other symbols. Quarter notes are notes with a note head and stem. Eight notes are notes with a note head and one flag or beam. Different types of quarter notes and eight notes can be seen in Figure 1.

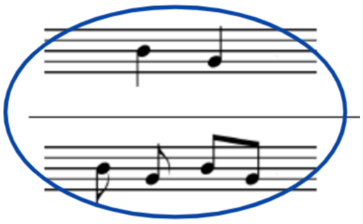

Figure 1: Quarter notes and eight notes.

It is also assumed that every staff bar begins with a G clef, see Figure 2.

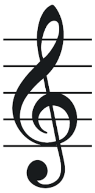

Figure 2: A G clef [3].

The system is also delimitated to recognize sheet music from scanned images only, since no correction for optical distortions is made.

2 Theory

Optical Music Recognition (OMR) is a form of image processing and a subcategory of Optical Character Recognition (OCR). It refers to the technique of optically interpreting sheet music into digital form. To get an idea of the characteristics of the symbols to be recognized and the challenges associated with OMR, a brief explanation of music notation is given below.

2.1 Music Notation

Music notation is sound represented with symbols. A basic element which is present in essentially all sheet music is the horizontal lines called The Staff. Each staff consists of five parallel lines, and the four spaces between them. Both the lines and spaces determine how the rest of the musical symbols should be interpreted.

A symbol which is usually present to the far left of the staff is the G-clef, see Figure 3. The G-clef indicates the certain kind of pitch of the notes on the staff. When a G-clef is present the five lines are represented by the notes: E G B D F from bottom up. The four spaces represent F A C E from the bottom up. [3]

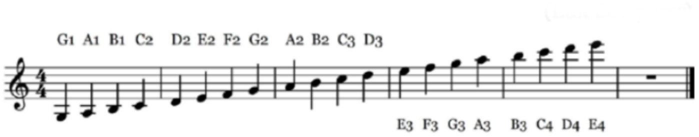

Figure 3: The Staff [3]

A note is the pitch and duration of a sound and can consist of three basic elements: a head, a stem and flags. If the note only consists of a head it is called a whole note and if the head is filled it's a half note. The placement of the head on the staff determines what note/pitch the note will have. The stem is the vertical line connected to the head and if attached to a whole note it makes a half note. If the note consists of a filled head and a steam it is a quarter note. The stem can point upwards or downwards from the head, but the direction has no effect on the note and is merely for visibility.

If there's a flag on the end of the stem it cuts the time value of a note in half, making a quarter note with a flag an eight note, and one with two flags a sixteenth note. If two or more notes containing flags are placed next to each other they are often grouped by so called beams. Beams are visualized as horizontal lines between the notes. If for example two eight notes are grouped with a beam their values can be summed and read as a quarter note, see figure 4.

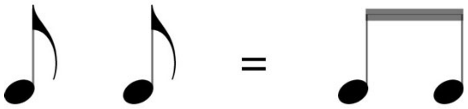

Figure 4: Beaming two eight notes makes a quarter note. [9].

The symbols discussed above are the most basic graphical elements that the

OMR will have to take into account when analyzing the test images. The test data will however contain images where other symbols are present, which is important to consider as well, as they should be ignored. The string output from the OMR will contain the individual notes, whether they are grouped by beams or not.

Due to the graphical properties of printed music as opposed to printed text the challenge of OMR is a different one than that of OCR. When working with text it is a one dimensional problem, since a line of text follows a horizontal baseline across a page and the characters can be interpreted sequentially along the x-axis. Music extends to a two dimensional problem, where the characteristics of one music note is dependent on it's vertical position as well as the horizontal [10]. As a result, the order in which to interpret the individual musical elements is not as obvious as in the one dimensional case. Another challenge of OMR is the fact that the same musical element can have more than one graphical representation. One example is the note containing a stem and flag, where the stem and flag could point upwards or downwards and still indicate the same note, as shown in Figure 3. In addition, some of the musical elements have similar graphical properties with only minor differences, making it potentially difficult to differentiate a symbol from another.

2.2 Mathematical Theory

The mathematical concepts that have been used in this project to detect staff lines, note heads and more are described in this section.

2.2.1 Hough Transform

Hough transform is a technique to detect lines and shapes in an image. Lines can be represented as shown in 1, where r is the length of a line perpendicular to the line and through origo, and thetha is the angle from r to the x-axis [1].

$$x \times cos(\theta) + y \times sin(\theta) = r \tag{1}$$

Every point in the image plane becomes a curve in the r, theta-plane and every line in the image-plane is represented by a point in the r, theta-plane, see Figure 5 [3].

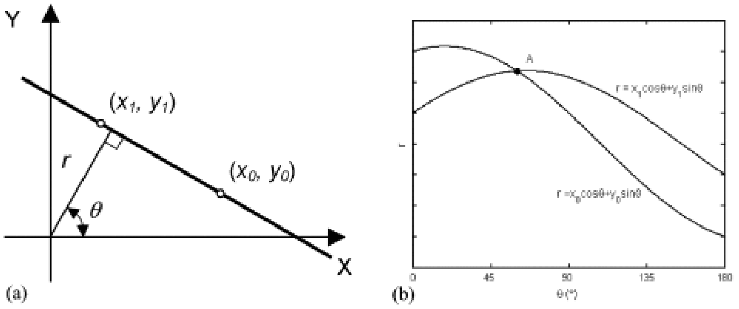

Figure 5: Hough transformation of a straight line in xy-plane to a single point in r, theta-plane [2].

Drawing every non-black pixel in the image as r and theta will result in curves in the r, theta-plane. These curves will intersect and points in the r, theta-plane where a lot of curves intersect indicates a line in the image [4].

2.2.2 Horizontal Projection

A horizontal projection calculates all pixel values horizontaly across the image, creating a histogram as seen in Figure 6.

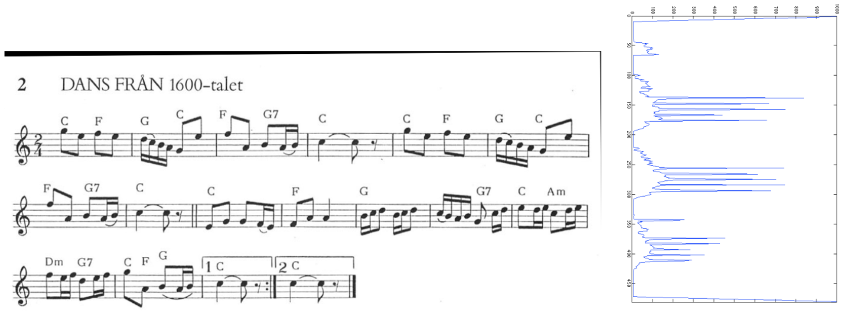

Figure 6: Example sheet music and the histogram created by horizontal projection [3].

In a binary image, greater values, or peaks, in the histogram indicates many white pixels in the associated row the image. Horizontal projection can be used to detect long lines in an image [3].

2.2.3 Morphological operations

Morphological operations takes a binary image and a so called structuring element as input and processes the image based on shapes. The structuring element is a small image with value 1 in certain pixels [3]. The structuring element is shifted over the image and is compared to the underlying pixel values using a defined operator [13]. If the two set of values match the condition defined by the operator, the pixels in the binary image is set to 0 or 1. In the implementation of the OMR the operators opening and closing is used throughout to shrink or grow different parts of the binary image.

Opening To perform opening of an image two different morphological operators are used: erosion followed by dilation. Erosion removes pixels on object boundaries while dilation adds pixels to the boundaries. The number of pixels added depends on the size and shape of the structuring element. Opening can be used to remove noise by smoothing contours, deleting thin connections between objects and eliminating small isolated objects [3], see figure 7.

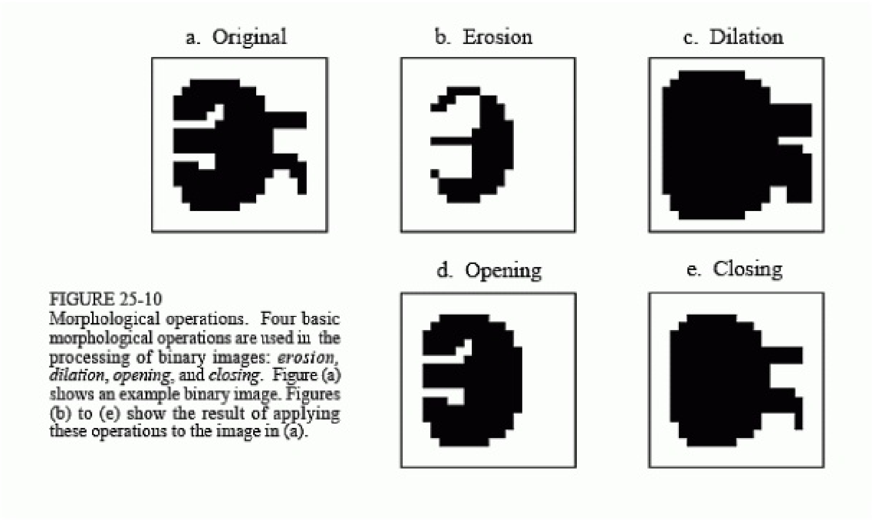

FIGURE 25-10
Morphological operations. Four basic morphological operations are used in the processing of binary images: *erosion, dilation, opening,* and *closing*. Figure (a) shows an example binary image. Figures (b) to (e) show the result of applying these operations to the image in (a).

Brainbitz

Figure 7: An example of the morphological erosion, dilation, opening and closing. [12].

Closing The closing of an image is defined by dilation followed by erosion. Closing can be used to remove small holes in an object as well as filling small cracks [13], as shown in figure 7.

2.2.4 Otsu's Thresholding Method

Otsu's method is used to automatically reduce a greylevel image to a binary image. The method finds an optimal threshold based on the observed distribution of pixel values. Otsu's chooses a threshold that minimizes the intraclass variance of the black and white pixels, where the intraclass variance describes the similarity between the pixels of the same class [11].

2.2.5 Template matching and cross-correlation

Template matching is a method to find objects in an image that matches an object in a template image. This can be done by using cross-correlation. The cross-correlation of two objects, f and h, is calculated as in 2, where m_2 is equal to m/2 and n_2 is equal to n/2 [3].

$$g(x,y) = \sum_{k=-n_2}^{n_2} \sum_{j=-m_2}^{m_2} h(j,k) f(x+j, y+k), \tag{2}$$

Greater correlation equals greater similarity of the image and template image.

3 Method/Implementation

The program is developed in Matlab R2018a. The input is images of scanned sheet music and the output is a string representing the detected quarter- and eight notes in the image.

3.1 Test input

To test the program eight images of scanned sheet music was used as input. An example of one of these is shown in Figure 8.

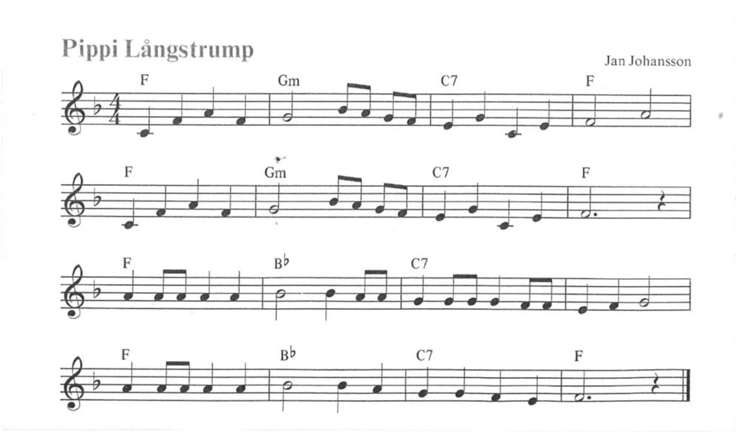

Figure 8: Sheet music to the song 'Pippi Långstrump' by Jan Johansson. [3]

3.2 Preprocessing

Before beginning the process of identifying the sheet music, the input image must be modified. First the image is converted to double precision and converted from rgb to gray scale using the Matlab functions `2double` and `rgb2gray`. Then the image is normalized so that the pixel values are between 0 and 1. This is calculated using 3.

$$newValue = Value - min(Value)/max(Value) \tag{3}$$

3.3 Binary Image

To be able to further process the image it is converted to a binary image. This is done by computing a global threshold of the image using the function `graythresh` and is then binarized with the function `imbinarize` using the threshold. The image is then inverted to make the background black and the objects white.

3.4 Rotation

It is desired to have the staff lines completely horizontal, however this is not always the case with scanned sheet music. This is fixed by using hough transformation to detect the staff lines and the angle it should be rotated around. The image is rotated with -90 degrees using the function `imrotate`. The hough transform is calculated using the matlab function `hough`. The strongest line is calculated and used to define the angle of which the image should be rotated.

The hough transform is first calculated on all angels with the precision of 1. The hough transform is then calculated again with a precision of 0.01 in the interval [-1,1] from the angle calculated in the first hough transform. The image is rotated with the calculated rotation angle and an additional 90 degrees to compensate for the -90 degrees rotation in the beginning.

The rotation of the image adds some extra pixels in the edges of the image. The number of added pixels on every side of the image is calculated using 4 and 5, and used to crop the image.

$$extraPixelsInY = tan(angle) \times widthOfRotatedImage \tag{4}$$

$$extraPixelsInX = tan(angle) \times heightOfRotatedImage \tag{5}$$

3.5 Staff Lines

An image containing only the staff lines is created using opening with a horizontal linear element with length 3 and then performing the morphological operation top hat, which is the image minus the opening of the image [5]. Thereafter, to restore the staff lines that have been damaged, a closing with a linear horizontal element with length 20 is performed and all connected objects of 15 pixels or less are removed.

The staff lines are detected using horizontal projection. The image is summed horizontally and the peaks with a value greater than 130 is stored. The first and last line of every staff bar is found by calculating the distance between every line, if the distance between two staff lines are more than 15 the lines are assumed to be belonging to two different staff bars.

To be able to determine the pitch of the note, the rows where the note heads centers could be located are calculated and stored. This includes the five detected lines and three additional lines above the first line and two under the last line, as well as the middle of two lines. In Figure 9 these pitch-rows are represented as green lines.

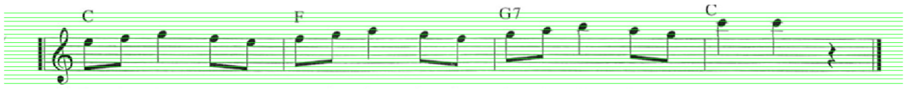

Figure 9: Rows where the center of the note heads could be placed, determining the pitch.

3.6 Segmentation

The image is segmented with every sub image containing a single staff bar. This is done by cropping the image from the first pitch-row to the last pitch-row with a margin the size of the space between the staff lines. The sub image is then scaled to a fixed height of 495 pixels. Scaling the image to a fixed height makes all equal objects, like note heads, approximately the same size, making it easier to recognize them. The pitch-rows are also scaled with the same scale factor.

As mentioned before, every staff bar starts with a G-clef. Since the G-clef is not to be detected it is cropped out of the image, to ensure that it is not mistaken for a note. This is done by template matching using normalized cross-correlation between the sub images and a template of a G-clef, see Figure 2. The template image is scaled to match the approximate size of the G-clefs in the sheet music, and made into a binary image.

An example of the sub image, binarized and the G-clef cropped out, can be seen in Figure 10.

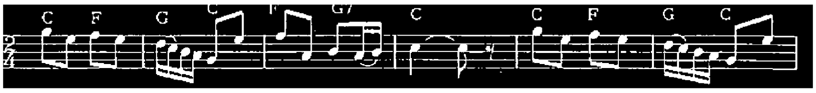

Figure 10: Binarized image of a staff bar, without the G-clef.

3.7 Detect note heads and remove undesired notes

To detect the note heads in the image, the staff lines are first removed. This is done using opening with a vertical linear element with length 15. Since the sixteenth notes are not to be detected, the notes with double beams or flags are identified and removed. This is done by first labeling 8-connected objects in the image using the function `bwlabel`. The function returns a matrix where every pixel that belongs to a certain region is represented by an integer value. [6]

The function `regionprops` was used to extract information about each region in the label matrix. Two set of properties where measured to detect the notes of interest. The first measurement returned a set of bounding boxes for each region, where a bounding box is the smallest rectangle containing the region. A second property uses Euler numbers to detect the number of holes in a region. The function then returns a scalar value for each region, representing the number of objects in the region minus the number of holes in those objects. [6] Since the regions containing notes with double beams or flags will always contain two holes, and the other notes of interest will contain less holes, the regions where the Euler numbers are less than zero where extracted. The bounding boxes containing the regions of interest where then applied to the sheet image simply making the pixel values in those regions black.

Opening with a disk-shaped element with radius 19 is then used to remove everything in the image but objects that could be note heads. The objects left are labeled and the center, width and height of the objects are stored in a table. An example of the resulting image can be seen in Figure 11, with the center of the disk-shaped objects marked.

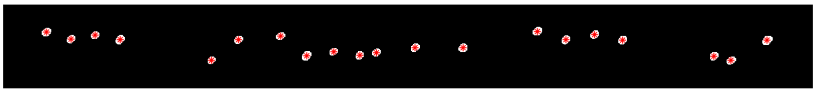

Figure 11: Staff bar with detected note heads and marked centers.

3.8 Determine time value and pitch

To determine the time value of a specific note, the flags and beams in the image are extracted and the number of flags or beams belonging to the note head is calculated. This is done by first removing the staff lines as described in section 3.7. Then the stems are found, by opening with a horizontal linear element with length 12, and subtracted from the image. The note heads are also subtracted from the image. An example of an image with the flags and beams extracted is shown in Figure 12.

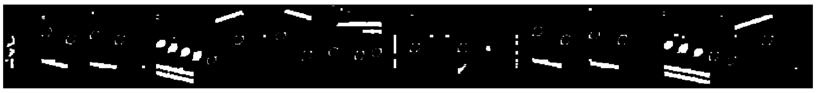

Figure 12: Staff bar with detected flags and beams.

To determine the number of flags or beams of every note, the area above and below every note head are scanned. Since the beams or flags are not always placed directly above the note heads, the width of the scanned are one width of the note head on every side of the note head. The number of objects with a size of 700 to 5000 are found and the largest number of objects above or below is stored. If a note have no objects above or below it, it is considered a quarter note and if the note has one object above or below it, it is considered an eight note. If there are more than 1 object detected the notes are ignored.

To determine the pitch of the notes the minimum distance to a pitch row is calculated. The output is a two character string of the encoding of the notes using upper case letters if it is a quarter note and lower case letters if it is an eight note. The character *n* is added to denote the end of a staff bar.

4 Results

The result is a matlab program that takes an image of scanned sheet music as input and outputs a string with the encoding of the notes, with upper case letters for quarter notes and lower case letters for eight notes. The example input image shown in Figure 8 gives the output string *"C2F2A2F2b2a2g2f2E2G2E2nC2F2A2 F2b2a2g2f2E2G2C2E2nA2a2a2A2A2B2a2a2G2g2g2G2f2f2E2F2nA2a2a2A2A2B2 A2G2G2F2E2"*. From the eight input images used to test the program, the worst accuracy was 74% correctly determined notes, and the best gave a 100% accuracy.

5 Discussion

5.1 Alternative Methods

5.1.1 Rotation

When compensating for the potentially rotated image the Hough Transform was applied to locate the horizontal lines. An alternative approach could be to use Horizontal Projection, which is one of the most widely used method to detect staff lines. [10] When performing a Horizontal Projection the pixels of the image is mapped to a histogram in which the staff lines will appear as distinct peaks. The histogram can however differ vastly when just small rotation angles are present in the scanned images. This was the major disadvantage taken into account when deciding to use the Hough Transform instead. Hough turned out to be more optimal for identifying the rotated lines since it can locate any straight lines regardless of direction. In addition, the Horizontal Projection method is only optimal for parallel lines. In sheet music, there are always other objects present on the staff, which may leave the lines deformed and disturb the projections. After rotating the entire image Horizontal Projection could be used on the subdivided images since the histogram of the 5 staff lines in each segment could easily be distinguished after compensating for the rotation. The Horizontal Projection is not as mathematically demanding, making it a better option when detecting the lines in the sometimes large set of individual staffs.

5.1.2 Segmentation

Another method to detect and divide every staff bar into a sub image is to use dilation to close the spaces between the staff lines. This results in that each staff bar becomes a complete filled object, see example in Figure 13. The bounding boxes of the staff bar objects can be detected and used to create sub images of every staff bar [8].

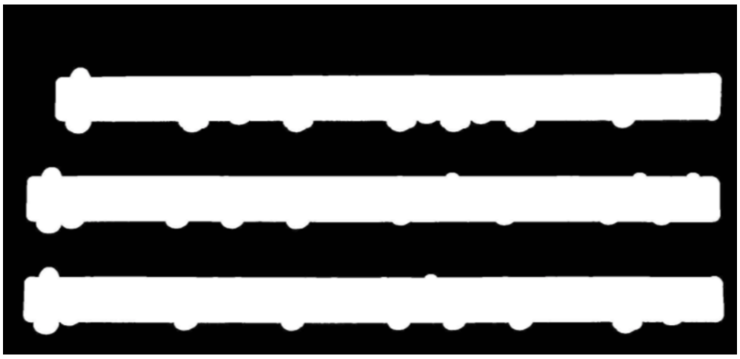

Figure 13: Dilation on image of three staff bars. [8]

This method was implemented and tested but gave a less accurate result than the method described in section 3.6 and was therefore not used.

5.2 Classification

The accuracy of the OMR is in many ways dependent on the classification of objects and distinguishing one musical element from another. This can be a challenge considering some musical elements possess similar graphical properties. One example were classification becomes challenging is if the morphological operations on the image remove or add more pixel values then desired, deforming the objects slightly. When determining what notes are of interest with the Euler number method it is assumed that the objects containing holes are indeed closed, since just a small crack in the outline of the hole will be interpreted as an object without holes. In the same way another object might have received additional holes from the morphological opening.

Aside from notes, the staff lines often contain multiple musical elements that the OMR shouldn't take into account and in a best case scenario the image would be cleaned up completely from such regions. This would not only make it easier to classify the desired notes but also make the labeling and classification more computationally efficient. It is however hard to remove all pieces of unnecessary information without losing relevant information as well.

5.3 Further Development

The program could be further developed by improving the methods so that the number of correctly identified notes increased. One way to improve this could be to remove more noise from the images. It would also be desirable to be able to read notes from photos of sheet music, where the images differ in for example lighting, perspective distortion, blurriness and optical distortion.

Another improvement would be to make it possible to define more notes and symbols on sheet music or being able to read hand written notes.

References

[1] R. Fisher, S. Perkins, A. Walker and E. Wolfart. *Hough Transform.* Image processing learning resources, 2003. http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm (Accessed: 20181214).

[2] A.A.Kassima, ZhuMiana, M.A.Mannanb. *Connectivity oriented fast Hough transform for tool wear monitoring.* ScienceDirect, 2004. https://www.sciencedirect.com/science/article/abs/pii/S0031320304000561 (Accessed: 20181214).

[3] D. Nyström. *TNM034 Advanced Image Processing.* Linköpings Universitet, 2018.

[4] U.Sinha. *The Hough Transform.* AI Shack. http://aishack.in/tutorials/hough-transform-basics/ (Accessed: 20181214).

[5] MathWorks. *bwmorph.* MathWorks. https://se.mathworks.com/help/images/ref/bwmorph.html (Accessed: 20181214).

[6] MathWorks. *bwlabel.* MathWorks. https://se.mathworks.com/help/images/ref/bwlabel.html (Accessed: 20181216).

[7] MathWorks. *regionprops.* MathWorks. https://se.mathworks.com/help/images/ref/regionprops.html (Accessed: 20181216).

[8] R. Lehman-Borer. *Optical Music Recognition.* Institutional Scholarship, 2016. https://scholarship.tricolib.brynmawr.edu/handle/10066/18782?show=full. (Accessed: 20181214)

[9] John Wiley and Sons, Inc. https://www.dummies.com/art-center/music/guitar/how-musical-notes-are-constructed/

[10] Kluwer Academic Publishers. *The Challenge of Optical Music Recognition* 2001. https://scholarship.tricolib.brynmawr.edu/handle/10066/18782?show=full. (Accessed: 20181214)

[11] Kluwer Academic Publishers. *Otsu's Method for Image Segmentation* 2017. https://medium.com/google-earth/otsus-method-for-image-segmentation-f5c48f405e (Accessed: 20181214)

[12] Aswin Pv. https://www.slideshare.net/aswinishere007/dilation-and-erosion-39733096

[13] R. Fisher, S. Perkins, A. Walker and E. Wolfart. *Morphology* 2003. http://homepages.inf.ed.ac.uk/rbf/HIPR2/morops.html (Accessed: 20181214)