Pay Attention to the Keys: Visual Piano Transcription Using Transformers

Uros Zivanovic[1] , **Ivan Pilkov**[2] and **Carlos Cancino-Chacón**[2]

[1]Universty of Trieste, Italy

[2]Institute of Computational Perception, Johannes Kepler University Linz, Austria

uros.zivanovic@studenti.units.it, ivan.pilkov@jku.at, carlos_eduardo.cancino_chacon@jku.at

Abstract

Visual piano transcription (VPT) is the task of obtaining a symbolic representation of a piano performance from visual information alone (e.g., from a top-down video of the piano keyboard). In this work we propose a VPT system based on the vision transformer (ViT), which surpasses previous methods based on convolutional neural networks (CNNs). Our system is trained on the newly introduced *R3* dataset, consisting of ca. 31 hours of synchronized video and MIDI recordings of piano performances. We additionally introduce an approach to predict note offsets, which has not been previously explored in this context. We show that our system outperforms the state-of-the-art on the PianoYT dataset for onset prediction and on the *R3* dataset for both onsets and offsets.

1 Introduction

Automatic music transcription (AMT) is the process of converting a music recording (typically an audio recording) into a symbolic representation (such as MIDI or sheet music) using computational methods [Benetos *et al.*, 2019]. AMT is a cross-modal task that bridges audio and symbolic music processing and is a fundamental task in music information retrieval (MIR). The generated symbolic music representation is often inspired by MIDI, with notes represented by their onset (start time), offset (release time), pitch, and velocity (loudness). Various MIR tasks, such as music search (indexing, recommendation), automatic music accompaniment, and music generation, benefit from the symbolic music representation produced by an AMT algorithm [Benetos *et al.*, 2019; Schedl *et al.*, 2014]. AMT systems also facilitate the creation of new datasets [Zhang *et al.*, 2022; Kong *et al.*, 2022], which can support a range of MIR tasks.

Although audio-based methods have been the primary focus of past AMT research, transcription from visual information, either alone or in combination with audio, presents an interesting approach to exploring cross-modal mappings, particularly in linking finely controlled gestures/movements with musical actions. Learning these mappings is valuable not only for AMT itself but also as an intermediate representation for tasks that connect motion to music-making, such

as music learning tools that could provide feedback during a performance.

In this work, we focus specifically on the case of automatic piano transcription (APT) for silent video. A top-down video can provide a substantial amount of information about a piano performance, allowing for the direct visualization of note onsets, offsets, and pitch-values based on key-presses and releases. Videos with such setup are a popular way to showcase piano-related content on platforms such as YouTube.[1] Research into video-based APT has shown that visual input can supplement audio-based methods, helping models handle harmonics [Wan *et al.*, 2015] and improving overall system performance in cases where the audio data is distorted [Koepke *et al.*, 2020]. Additionally, video can assist with tasks such as video–audio synchronization and it can provide a transcription solution in scenarios where no audio is available for a musical performance.

Previous research on visual piano transcription (VPT) has used convolutional neural networks (CNNs) for this task [Wang *et al.*, 2021; Koepke *et al.*, 2020; Wan *et al.*, 2015]. Recently, the video vision transformer (ViViT) [Arnab *et al.*, 2021] has been shown to be competitive with state-of-the-art (SotA) CNNs in computer vision benchmarks. This is leading to a shift in the computer vision space from CNNs to transformers. Training a machine learning VPT system requires data in the form of videos and corresponding MIDI files. Current machine learning methods rely on datasets generated from audio-based APT for training and testing (i.e., using APT to transcribe the audio track in the video, and use the resulting MIDI file as the ground truth) [Su *et al.*, 2020; Koepke *et al.*, 2020]. However, using these datasets may introduce artifacts due to limitations inherent in audio-based AMT. In particular, recent work has highlighted the limitations of APT, primarily due to SotA APT methods being trained on the same dataset (the MAESTRO dataset [Hawthorne *et al.*, 2019]). Additionally, audio-based APT is limited in accurately transcribing note durations for this purpose, due to pianists' use of the sustain pedal. [Hu *et al.*, 2024; Marták *et al.*, 2024; Edwards *et al.*, 2024]. Furthermore, current VPT methods focus on predicting onsets and pitches only [Koepke *et al.*, 2020; Suteparuk, 2014], but not offsets (with the exception of [Su *et al.*, 2020], which gener-

---

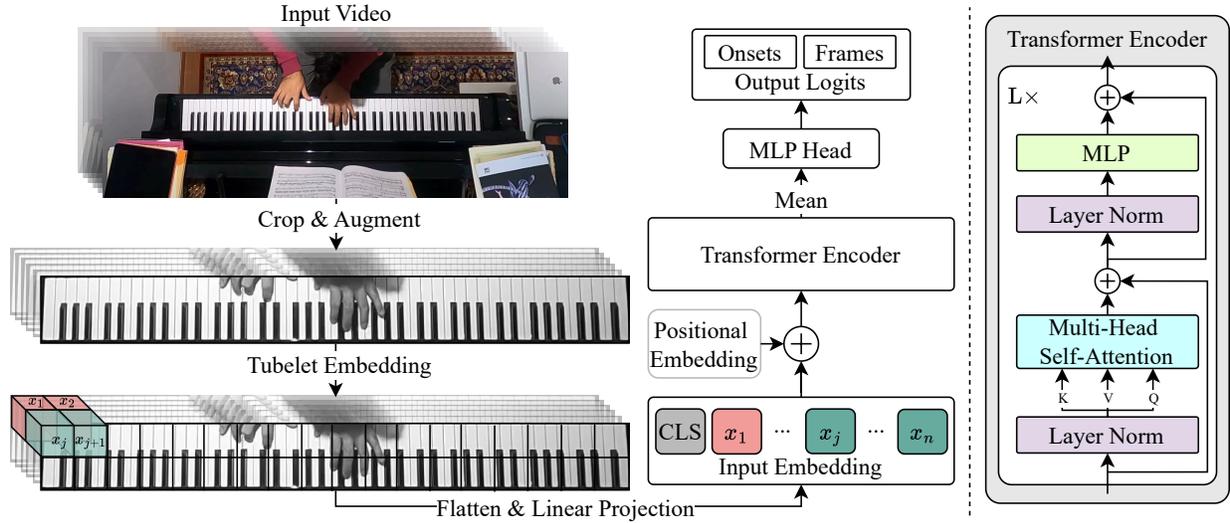[1]See, e.g., https://www.youtube.com/watch?v=xspj9uE5TQQ

Figure 1: Overview of the model architecture.

ates an intermediate piano-roll–like representation to generate audio).

The contributions of this paper are the following:

1. We propose a system based on the vision transformer (ViT) [Dosovitskiy *et al.*, 2021] that is competitive with the SotA in VPT. The proposed system is illustrated in Figure 1.

2. We investigate two approaches to predict offsets, namely predicting frames directly, as well as an approach based on the Onsets and Frames (OaF) APT model for audio [Hawthorne *et al.*, 2018] and we compare the overall performance of the two for the purpose of our task.

3. We introduce the *R3* dataset, containing 895 synchronized and high quality video, audio, and MIDI recordings of two professionally trained pianists practicing various pieces, amounting to ca. 31 hours of music. To the best of our knowledge, this is the largest dataset for the task, being almost twice the size as the previous largest dataset [Koepke *et al.*, 2020] (which consists of transcribed piano performances, not real "performed" MIDI files).

Alongside our contributions, we also present the rationale behind key choices made during the design of our system. These include the results of our experimentation in progressively adding data augmentation techniques when training the ViT, as well as our use of loss weights to deal with the sparsity of positive note events. Lastly, we present the overall performance of our model and compare it directly to other SotA methods.

The structure of this work is as follows: Section 2 reviews related work on visual transcription and multi-modal computer vision tasks. Section 3 describes our ViT-based transcription architecture. Section 4 describes the R3 dataset. Section 5 describes the experimental setup. Section 6 presents the results of our experiments. Section 7 provides a discussion. Finally, Section 8 concludes the paper.

2 Related Work

In this section we discuss some related work on APT and transformer-based multi-modal computer vision tasks.

2.1 Audio-based Automatic Piano Transcription

Piano transcription is one of the most popular subtasks in AMT. Its popularity is partially due to the relatively limited number of degrees of freedom for transcribing piano (i.e., needing onset and offset times, played note and potentially pedal), when compared with other instruments like violin or voice. Furthermore, there are large datasets for audio-based piano transcription, including the MAESTRO dataset [Hawthorne *et al.*, 2019]. A comprehensive review of AMT is provided in [Benetos *et al.*, 2019].

Early approaches for audio-based APT used a sliding window over the audio spectrogram, predicting whether each note was present in the current "frame" [Kelz *et al.*, 2016]. Such an approach is similar to the one taken by VPT models, which iterate over the frames in a video [Koepke *et al.*, 2020; Suteparuk, 2014]. The OaF model [Hawthorne *et al.*, 2018] was the first audio based AMT model to conduct dual-objective piano transcription, showing that it can be beneficial to model note onsets separately from frames.

CNNs have been the model of choice for spectrogram-based APT [Hawthorne *et al.*, 2018; Kelz *et al.*, 2016; Kong *et al.*, 2022]. In recent years, however, transformer-based models been shown to work well for both multi-track transcription [Gardner *et al.*, 2022] and single-track piano transcription [Hawthorne *et al.*, 2021]. By using end-to-end machine learning architectures, these works move the problem from hand designing a system for AMT to creating a high-quality dataset and fitting a model to it.

2.2 Video-based Automatic Piano Transcription

Because of the easily discernible black and white keys of the piano, traditional computer vision algorithms are able to register a piano keyboard in an image and even identify separate

keys. This enables such tools to detect onsets on the piano with a high degree of accuracy [Suteparuk, 2014]. Such an approach is very difficult to apply to various lighting and environmental conditions, however. It also struggles in cases where the piano keys may be slightly obscured, or when the piano is at an angle in the image. Additionally, traditional computer vision tools are unable to capture and reason about complex movements across multiple frames in a video.

More recently, CNNs have been shown to work well for VPT. By using a modified ResNet [He *et al.*, 2016] model, [Koepke *et al.*, 2020; Su *et al.*, 2020] were able to make onset and pitch predictions directly on a cropped image of the piano. [Koepke *et al.*, 2020] showed that their vision-based model can compensate for poor quality audio, helping an audio-based AMT algorithm perform better. There are also systems that are explicitly designed around using both video and audio simultaneously. Such an audio-visual model is presented by [Wang *et al.*, 2021]. In their approach, they use the video model primarily for eliminating harmonic errors. This is done by tracking the pianists hands and eliminating any predicted notes that would be impossible to play from that position. [Su *et al.*, 2020] present Audeo, an approach that generates audio by first generating an intermediate pianoroll-like representation, followed by a synthesis step. The main component of this intermediate step, referred to as Video2Roll Net, is a ResNet model similar to the one used by [Koepke *et al.*, 2020], but enhanced with feature attention for improving visual cues at multiple scales.

2.3 ViT for Video and Multi-modal Applications

Since ViT showed that transformer architectures can outperform comparative CNNs in the ImageNet benchmark [Russakovsky *et al.*, 2015], there has been an increased interest in transformers for computer vision. When adapting ViT to video, there have been numerous architectures proposed, although most focus on classification. SwinBERT [Lin *et al.*, 2022] is a multi-modal architecture which is able to take a video clip and generate a text caption describing it. While SwinBERT uses both a Swin transformer [Liu *et al.*, 2021] and a multimodal transformer encoder to generate the caption text, we are able to avoid needing a second model thanks to the simplicity of piano transcription compared to language.

Our proposed model is instead more similar to the spatio-temporal attention variant of ViViT [Arnab *et al.*, 2021; Bertasius *et al.*, 2021]. This type of ViViT is a direct generalization of ViT to video, and has been utilized by the Video Masked Autoencoders [Tong *et al.*, 2022] for a variety of video classification tasks including fine-action detection. The Video-Audio-Text transformer [Akbari *et al.*, 2021] also utilizes the same method, but appends additional audio and text tokens to the input sequence to conduct multimodal self-supervised learning, generating representations that are useful for both video and audio classification.

3 Visual Transcription Architecture

In this section we describe the details of our ViT-based transcription model. For a visual overview of the system, see Figure 1.
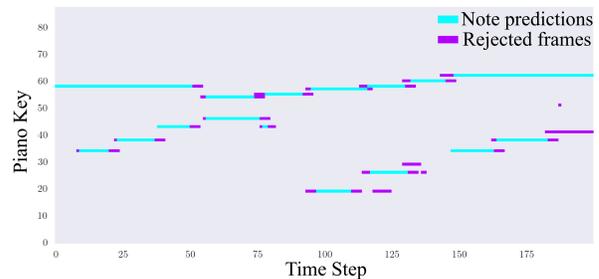

Figure 2: An illustration of accepted and rejected frame predictions.

3.1 Predicting Onsets and Offsets

When considering methods for onset/offset prediction, the simplest choice is to conduct 88 classifications for each frame in the video, predicting what notes are present in each given frame. These framewise predictions can then be concatenated into an array to form a prediction for the entire video. A VPT method that utilizes this approach is the Audeo model [Su *et al.*, 2020], which additionally places an extra generative adversarial network on top of the video predictions.

Splitting a note up into distinct onset and frame predictions, enables us to reject spurious positive frames when they are missing a corresponding onset. An example of this is visualized in Figure 2. Modeling onsets separately from the rest of the note is also closer to what is happening physically in the video, as pressing down a note and holding down a note are two different actions.

Inference. When performing inference over an entire video, we iterate over all frames and then concatenate the predictions to produce an array of shape

$$(88 \times 2, \texttt{number\_of\_frames} - \lceil \texttt{window\_size}/2 \rceil \times 2).$$

Some frame predictions are missing at the start and end because we need enough frames to create a window. An alternative to skipping the first and last few frames could be to pad the start and end of the video. As there are 88 notes, to predict both onsets and frames we require 176 total predictions per frame. Because we train our model on a temporal resolution of $30\,\text{fps}$, all predicted notes get assigned to their nearest frame, leading to a potential error of $15\,\text{ms}$ even given perfect predictions. Achieving a high temporal accuracy is therefore a key area where VPT is more challenging than audio based AMT.

3.2 Post-Processing

After iterating through the video, the resulting array still contains some noise. To remove this we apply a Gaussian filter along the time axis (similarly to [Koepke *et al.*, 2020]) individually to the onset and frame predictions. For onsets we use $\sigma = 0.2$ and a kernel radius of 8, while for frames we use $\sigma = 0.8$ and a kernel radius of 4. We then threshold the smoothed array using a value of 0.5.

To identify onsets, we find regions in the onset predictions which are positive for 1 or more consecutive time steps, and assign the corresponding onset to the time step with the highest predicted value in that region. After finding an onset for a

note, we then iterate over all positive frames that succeed that onset until we reach the end of the predicted frames. The final 4 frames are ignored in order to compensate for the Gaussian smoothing (which can cause false positives up to the size of the kernel radius). At this point we have a completed piano roll, which can be converted directly into MIDI. Although our model outputs both onsets and frames, we can also choose to utilize either one of them to predict only onsets or only frames. We take this approach in Section 6.2 to evaluate how well frame predictions perform for predicting onsets and offsets.

3.3 Vision Transformer Backbone

We introduce a new backbone architecture for VPT built upon the ViT, which accepts a video clip sampled at $30\,\mathrm{fps}$ consisting of 6 frames with a resolution of $(64, 784)$, and outputs a vector of size (176) containing onset and frame predictions for the third frame in the input window. The architecture is visualized in Figure 1.

In order to convert the input video clip to embeddings, we utilize the tubelet embedding method used in ViViT [Arnab *et al.*, 2021]. Tubelet embeddings are a direct generalization to 3D of the "patchification" procedure used in ViT. Defining a tubelet size of $(t \times h \times w)$, we reshape a video with size (T, H, W) to:

$$\left(\left\lfloor \frac{T}{t} \right\rfloor \times \left\lfloor \frac{H}{h} \right\rfloor \times \left\lfloor \frac{W}{w} \right\rfloor, t \times h \times w \right).$$

A linear layer is then applied to project the flattened tubelets $x_i \in \mathbb{R}^{t \times h \times w}$ to $z_i \in \mathbb{R}^d$, where d is the embedding dimension. After appending a learned `[CLS]` token to the start of the sequence and adding the positional embeddings, the tokens are passed through a standard transformer encoder. While VPT approaches utilizing ResNets benefit from the addition of a "slope vector" to help the model identify the location of keys in the image [Koepke *et al.*, 2020; Su *et al.*, 2020], because ViT uses absolute positional embeddings, we do not require any such modifications.

As vision transformers have been shown to require extensive pre-training to achieve satisfactory results, we choose to utilize the high-quality distilled ViT-small weights provided by [Wang *et al.*, 2023]. The ViT-small model consists of 12 Transformer Encoder layers, an embedding dimension d of 384, 6 attention heads, and an MLP hidden dimension of 1536. The model was pre-trained on 16 frame long video clips with a resolution of (224×224) and a tubelet size of $(2 \times 16 \times 16)$. We change the resolution of the model to $(64, 784)$, and number of frames to 6. Although this is a large change, we choose the resolution such that the number of tokens per frame stays the same:

$$\frac{H}{h} \times \frac{W}{w} = \frac{64}{16} \times \frac{784}{16} = \frac{224}{16} \times \frac{224}{16} = 196.$$

When adapting the positional embeddings to the new number of frames, we simply remove the section that used to correspond to the extra frames and keep the rest. To generate onset and frame predictions, we follow the same procedure used during pre-training, taking the mean of the output embeddings and placing a linear head on top. Therefore, we are able

Dataset	No. Notes		Total Duration	
	Test	Train	Test	Train
R3s	98,045	292,814	2.9h	8.5h
R3x	119,725	479,629	3.6h	15.8h
R3 (Total)	217,770	772,443	6.6h	24.3h
PianoYT	33,535	369,922	1.7h	17.2h

Table 1: Statistics on the number of notes and total video duration in the R3 (R3s + R3x) and PianoYT datasets.
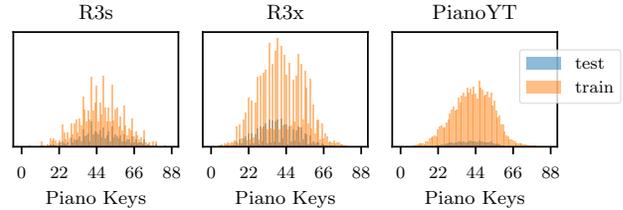

Figure 3: Note distribution for the R3 and PianoYT datasets.

to utilize the pre-trained model with a different resolution and number of frames without requiring the re-initialization of any weights or rescaling of the positional embedding.

4 Datasets

4.1 R3

In order to train our ViT-based model, we introduce a dataset consisting of ca. 31 hours (shown in Table 1) of recordings from 62 different piano practice sessions by two professionally trained pianists (each with an average of 14 years of formal music education). The recordings come from work on the *Rach3* dataset [Cancino-Chacón and Pilkov, 2024] with additional processing and annotations added for the purposes of the current work. Each session contains a top-down video with synchronized audio, high-quality FLAC, and MIDI recordings. We do not use the audio recordings in our experiments, but they are available for future work and for the benefit of the community. We also provide tight crops around the piano keys for all videos. Two different pianos are used throughout the practice sessions: a Yamaha GB1k Silent and an Essex EUP-116 equipped with a silent system (allowing for MIDI capturing). For audio we use AKG P170 condenser microphones and a Focusrite Scarlett 2i2 audio interface. The video is captured with GoPro cameras.

R3 includes two sub-splits which differ in content variety and size. The R3s split contains practice sessions from one pianist practicing Rachmaninoff's Piano Concerto No. 3 Op. 30 (and hence the name of the dataset), while R3x contains sessions from both pianists practicing a variety of pieces from the Western Classical repertoire. Both splits have been processed exactly the same way and differ only in musical content.

As part of our dataset preparation pipeline, we split all recordings into smaller chunks between ~ 50 seconds and

~ 447 seconds. This results in 895 individual sets of synchronized video, FLAC, and MIDI recordings. To create crops around the piano keyboard, we employ an automatic template matching approach with manual adjustments where necessary. We create train and test sets using a 25/75 split, resulting in 181 samples in the combined test and 714 samples in the combined train set. To ensure that each test set is comprised of completely different practice sessions from those present in the train sets, we ensure that no samples share the same date across splits. Figure 3 shows the distribution of MIDI pitches throughout the datasets.

4.2 PianoYT

Introduced by [Koepke *et al.*, 2020], this dataset consists of ca. 19.2 hours (shown in Table 1) of top-down piano performance videos downloaded from YouTube. Dataset labels are generated automatically using the OaF model [Hawthorne *et al.*, 2018]. We note that our version of the PianoYT dataset is missing 2 videos in the train set due to them no longer being available on YouTube.

The amount of notes and the total duration of both datasets are shown in Table 1. Our R3 dataset has not only a longer total duration than PianoYT, but the total number of notes is more than double, indicating a higher note density than PianoYT. This could be attributed to either faster piano pieces being played or less silence throughout the R3 dataset. PianoYT also has a significantly smaller test set than R3.

5 Experimental Setup

5.1 Video Pre-Processing

In order to make data loading easier and have a consistent temporal resolution throughout all datasets, all videos are first transcoded from their native frame-rate to a frame-rate of 30 using FFmpeg.[2] We then crop out the piano using the crops provided with the datasets. Lastly, we ensure that the image is in the correct orientation and is mirrored correctly. Specifically, this requires rotating all PianoYT images 180 degrees to have the same orientation as R3.

Video Augmentations: As data augmentations we randomly apply Gaussian noise, random rotations, brightness and scale jittering. We also utilize a grayscale transformation to remove potentially unimportant information from the image. Lastly, we normalize all frames using the mean and std. from ImageNet [Deng *et al.*, 2009].

5.2 MIDI Pre-Processing

When adding onset labels, we first assign each note onset to the nearest video-frame, giving it a label of 1. Then we additionally assign a value of 0.5 as an onset label for the next and previous video-frames. To calculate frame labels, we assign a label of 1 when the note is pressed down, with the video-frame before and after the positive region being given a label of 0.5. The additional positive values are added in order to increase the number of positive onset and frame values in the dataset and also to deal with possible differences in timing between datasets.

---

[2]https://ffmpeg.org/

5.3 Class Weights

We employ the binary cross-entropy (BCE) loss function, modeling our problem similarly to multi-label classification and conducting 176 binary classifications. When training for joint onset and frame prediction, we obtain the final loss by summing the loss from both predictions. Due to the sparsity of positive classes in the datasets, we also apply class weights directly to the output of the loss function, which gives us the ability to control the weight of the positive samples in the dataset. This results in the following loss function:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{N}\sum_{i=1}^{N} w_i \left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \right].$$

where y_i is the ground truth, $\hat{y}_i$ is the model output, and w_i is the weight for the i-th sample, and N is the number of samples. By changing the positive class weights, we can balance the precision and recall of the model.

When choosing what class weights to use, we first calculate the ratio of positive to negative frames for each note. We then rescale this to be within $[2, 4]$ for frames and $[3, 5]$ for onsets. Rescaling is something we found necessary, as we found very-high class weights led to unstable training. Because onsets are sparser than frames, we use larger positive class weights for them.

5.4 Training

We utilize either two GPUs with a total batch size of 32 or 4 GPUs with a total batch size of 384 when training on PianoYT or R3 respectively. When training on R3 we use the AdamW [Loshchilov and Hutter, 2017] optimizer with a weight decay of 0.05. When fine-tuning a model trained on R3 on PianoYT, we use SGD with a momentum of 0.9. The learning rate for all runs follows a cosine decay function with a warmup of either 10% or 25% of all training steps depending on whether we are training on PianoYT or R3 respectively. We utilize the same image augmentations as described in Section 5.1 across all runs on R3 and PianoYT.

5.5 Evaluation

We follow the standard evaluation practices used in AMT [Hawthorne *et al.*, 2018; Koepke *et al.*, 2020; Hu *et al.*, 2024; Ycart *et al.*, 2020] and we employ the precision, recall, and F-measure as our evaluation metrics. These metrics can be computed either *framewise*, where we consider each frame as a separate prediction, or *notewise*, where each note is described by its onset, offset, and pitch. Additionally, notewise metrics can be computed *onsetwise*, where we only consider the pitch and onset of each note, ignoring the offset, which is useful for comparing with previous work that does not predict offsets.

For onsetwise and notewise metrics, we evaluate model performance using the transcription evaluation utilities provided by `mir_eval` [Raffel *et al.*, 2014]. This implementation matches as many notes as possible by checking if the distance between predicted note onset/pitch and reference note onset/pitch is less than set tolerance values, before computing the respective metrics using this matching. We keep all

Configuration	F-score		
	Framewise	Onsetwise	Notewise
Color inputs	0.08	0.18	0.06
+ Grayscale	0.12	0.25	0.07
+ DataAugment	0.16	0.39	0.13
+ Class Weights	0.31	0.63	0.20

Table 2: The effect of progressively adding augmentations and class weights on ViT performance.

settings at their defaults except the onset tolerance, which we increase from $50\,\text{ms}$ to $100\,\text{ms}$. This corresponds to an increase from an onset tolerance of $\sim \pm 1$ to ± 3 frames, meaning that a predicted onset up to 3 frames away from the true onset would now be considered correct. This window size in the $100\,\text{ms}$ case is close to the limit of human detectability thresholds [ITU-R, 1998]. We show the effects different onset tolerances have on the model performance in Figure 4. For framewise metrics, we follow [Ycart *et al.*, 2020; Hu *et al.*, 2024], and we use a frame-rate of $30\,\text{fps}$ to match the frame-rate of the input videos.

6 Experimental Results

6.1 Augmentations and Class Weights

In order to explore the contribution of each design choice leading up to our final proposed ViT model training procedure, we perform an ablation study on the smaller R3s dataset. To show the effect of preprocessing, data augmentation techniques, and class weights, we iteratively add them to the training recipe and observe the change in the models performance on the R3s validation set after training for 5 epochs. More details on the exact hyperparameters used for the ablation study can be found in Appendix A in the supplementary materials.

Utilizing grayscale inputs already leads to an improvement in the model's performance, as is shown in Table 2. Adding the collection of data augmentation techniques discussed in Section 5.1, indicated as DataAugment, as well as the class weights discussed in Section 5.3, continued to improve metrics greatly. Overall, these results support our choice to use these techniques when training the ViT.

6.2 Predictive Accuracy

Here we show the performance of our ViT-based model on the R3 and PianoYT datasets and compare it to the previous ResNet-based models proposed by [Koepke *et al.*, 2020; Su *et al.*, 2020]. We also investigate the differences between using the onsets-and-frames approach for note predictions versus the frames-only one to see which could be a better modeling strategy for this task. This is done by training all models on joint onsets and frames, and then ignoring the onset predictions during post-processing. All results are shown in Table 3. The model called S2S is our re-implementation of ResNet with aggregation and slope module presented by [Koepke *et al.*, 2020], and V2R is the official implementation provided by [Su *et al.*, 2020] for their Video2Roll Net.

R3: We train all models on joint onsets and frames on R3 for 10 epochs, utilizing all the methods tested in Section 6.1. When training our ViT model we additionally find it necessary to use stochastic depth [Huang *et al.*, 2016] to combat overfitting. When training the CNNs we did not encounter overfitting to the same degree as with the ViT, and instead chose to use use early stopping. Additional details on the parameters for these runs can be found in Appendix B in the supplementary material. We evaluate metrics on the two subsplits of R3, R3s and R3x, as well as on PianoYT, which is not included in the training data.

In all metrics, our ViT model outperforms the other two. Interestingly, when looking at the performance on PianoYT, the ViT is the only model that somewhat generalizes to the unseen dataset. We also find interesting that, although there are some architectural differences between the two CNNs, they both perform very similarly across a wide range of metrics. This suggests that the most important factor for the performance of these models is the shared ResNet backbone.

When comparing metrics between onsets-and-frames and frames-only prediction approaches, the first approach seems to fall behind. Although onsetwise metrics are very similar for the two approaches, framewise and notewise metrics are higher when predicting only frames. Framewise scores seem to benefit the most when utilizing frame-only predictions.

PianoYT: When training on PianoYT, we utilize the models that were already pre-trained on R3 and then fine-tune them on PianoYT. For this reason, we find it sufficient to train for only 5 epochs on PianoYT. Detailed hyper-parameters for these runs can be found in Appendix B in the supplementary material. Since the PianoYT ground truth offsets are of low quality, we only train on onsets. Our ViT model shows better performance again, while both ResNet-based models achieve the same onsetwise scores. Perhaps the largest difference between the ViT and ResNet models is the ability of the ViT to better maintain performance on the R3 dataset it was pre-trained on.

7 Discussion

Our results across all datasets show strong scores in favor of the ViT based model. Our model is able not only to outperform previous ResNet based approaches on the data it was trained on, but also shows very promising generalization capabilities. We believe that these capabilities are the result of a combination of high quality pre-trained weights and a larger model that is able to learn and retain more information. The large size of R3 is also likely more beneficial to a large model that has the capacity to learn from the extra data.

We also discovered that a frames-only approach works better than joint onsets and frames. This could be due to the sparsity of onset events as compared to frame-level events, which could lead to poorer quality onset predictions. These results show a stark difference from audio based models, which have been shown to benefit from the joint modeling task. We did find the onset predictions useful when fine-tuning on PianoYT however, as they allowed us to adapt the joint onset and frame model to just onsets without having to make any changes.

| Model | R3s | | | R3x | | | PianoYT |
	Framewise	Onsetwise	Notewise	Framewise	Onsetwise	Notewise	Onsetwise
Trained on R3							
Predicting Onsets and Frames							
S2S [Koepke *et al.*, 2020]	0.31	0.84	0.18	0.46	0.83	0.46	0.06
V2R [Su *et al.*, 2020]	0.31	0.84	0.18	0.45	0.82	0.47	0.01
ViT (ours)	0.32	0.87	0.19	0.48	0.85	0.49	0.40
Predicting Frames only							
S2S [Koepke *et al.*, 2020]	0.51	0.82	0.27	0.63	0.81	0.55	0.24
V2R [Su *et al.*, 2020]	0.52	0.86	0.28	0.63	0.83	0.57	0.05
ViT (ours)	0.52	0.88	0.29	0.63	0.85	0.57	0.35
Trained on R3, finetuned on PianoYT							
Predicting Onsets Only							
S2S [Koepke *et al.*, 2020]	-	0.21	-	-	0.30	-	0.64
V2R [Su *et al.*, 2020]	-	0.21	-	-	0.30	-	0.64
ViT (ours)	-	0.58	-	-	0.72	-	0.68

Table 3: Performance comparison (F-score) of models trained on R3 and fine-tuned on PianoYT
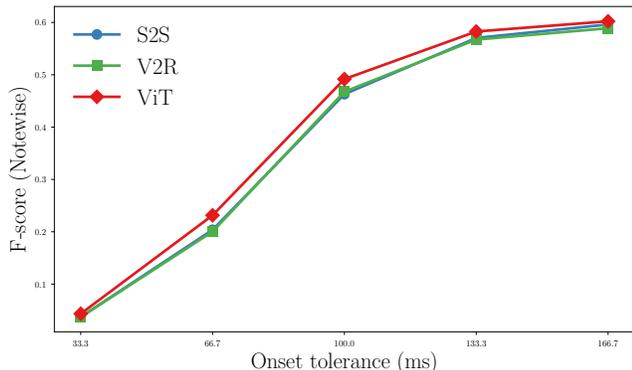

Figure 4: Notewise F-score on R3x for different onset tolerances for ViT, S2S and V2R models.

Onset Tolerances. To investigate the temporal accuracy of the models on the R3x dataset, we plot the notewise F-score in Figure 4. Although all models struggle to predict notes within a one frame tolerance, they get much better up to a 3 frame tolerance, at which point they start to plateau. These results show how challenging it is to make temporally accurate predictions at a frame-rate of $30\,\mathrm{fps}$, and highlight an area where VPT algorithms can still be greatly improved.

Supplementary Materials. We provide a Google Colab showcasing the inference procedure of our model and a YouTube playlist with example predictions at this web page.[3] Dataset, model weights, and training/evaluation code are also public and made available.

8 Conclusions

This paper introduces a transformer architecture for visual piano transcription. Our experimental results show that the proposed architecture can compete with and even outperform previous CNN-based models. To further aid model training, we introduce a new dataset, R3, containing synchronized video, audio and MIDI data. Additionally, we investigate the effect of positive class weights on model performance, highlighting their importance and impact on the final model. We also compare two approaches for note onset and offset prediction and discover that the frames-only approach could provide better results than the joint onsets-and-frames one for VPT in contrast to audio-based AMT, highlighting the differences between the two tasks.

Future research could focus on improving temporal accuracy by training on a higher frame rate or by making sub-frame predictions. Another possible direction is addressing the strong influence of the model's pre-trained weights on our method, potentially by conducting pre-training specifically for the VPT task. We would also like to investigate explicitly learning offsets, similarly to the approach used for onsets, to obtain better performance in notewise and framewise metrics. Furthermore, a full transformer Encoder–Decoder could be used to predict MIDI-like tokens directly from the video clip, using a larger temporal size, such as 32 frames, to process approximately one second of video at a time. This approach would function similarly to [Hawthorne *et al.*, 2021] and eliminate the need for a hand-crafted post-processing pipeline. Finally, more work in the direction of utilizing both audio and video for AMT could lead to overcoming the limitations of each of the modalities and achieving even better results.

---

[3]https://chromeilion.github.io/onf_vpt/

Acknowledgments

References

[Akbari *et al.*, 2021] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. VATT: transformers for multimodal self-supervised learning from raw video, audio and text. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 24206–24221, 2021.

[Arnab *et al.*, 2021] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. Vivit: A video vision transformer. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 6816–6826. IEEE, 2021.

[Benetos *et al.*, 2019] Emmanouil Benetos, Simon Dixon, Zhiyao Duan, and Sebastian Ewert. Automatic music transcription: An overview. *IEEE Signal Processing Magazine*, 36(1):20–30, 2019.

[Bertasius *et al.*, 2021] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 813–824. PMLR, 2021.

[Cancino-Chacón and Pilkov, 2024] Carlos Eduardo Cancino-Chacón and Ivan Pilkov. The rach3 dataset: Towards data-driven analysis of piano performance rehearsal. In Stevan Rudinac, Alan Hanjalic, Cynthia C. S. Liem, Marcel Worring, Björnthór Jónsson, Bei Liu, and Yoko Yamakata, editors, *MultiMedia Modeling - 30th International Conference, MMM 2024, Amsterdam, The Netherlands, January 29 - February 2, 2024, Proceedings, Part V*, volume 14565 of *Lecture Notes in Computer Science*, pages 28–41. Springer, 2024.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[Edwards *et al.*, 2024] Drew Edwards, Simon Dixon, Emmanouil Benetos, Akira Maezawa, and Yuta Kusaka. A data-driven analysis of robust automatic piano transcription. *IEEE Signal Processing Letters*, 31:681–685, 2024.

[Gardner *et al.*, 2022] Josh Gardner, Ian Simon, Ethan Manilow, Curtis Glenn-Macway Hawthorne, and Jesse Engel. Mt3: Multi-task multitrack music transcription. In *ICLR 2022*, 2022.

[Hawthorne *et al.*, 2018] Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse H. Engel, Sageev Oore, and Douglas Eck. Onsets and frames: Dual-objective piano transcription. In Emilia Gómez, Xiao Hu, Eric Humphrey, and Emmanouil Benetos, editors, *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*, pages 50–57, 2018.

[Hawthorne *et al.*, 2019] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *International Conference on Learning Representations*, 2019.

[Hawthorne *et al.*, 2021] Curtis Hawthorne, Ian Simon, Rigel Swavely, Ethan Manilow, and Jesse H. Engel. Sequence-to-sequence piano transcription with transformers. In Jin Ha Lee, Alexander Lerch, Zhiyao Duan, Juhan Nam, Preeti Rao, Peter van Kranenburg, and Ajay Srinivasamurthy, editors, *Proceedings of the 22nd International Society for Music Information Retrieval Conference, ISMIR 2021, Online, November 7-12, 2021*, pages 246–253, 2021.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[Hu *et al.*, 2024] Patricia Hu, Lukáš Samuel Marták, Carlos Cancino-Chacón, and Gerhard Widmer. Towards Musically Informed Evaluation of Piano Transcription Models. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2024.

[Huang *et al.*, 2016] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q. Weinberger. Deep networks with stochastic depth. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV*, volume 9908 of *Lecture Notes in Computer Science*, pages 646–661. Springer, 2016.

[ITU-R, 1998] ITU-R. Relative timing of sound and vision for broadcasting. Recommendation BT.1359, International

Telecommunication Union, Radiocommunication Sector (ITU-R), 1998.

[Kelz *et al.*, 2016] Rainer Kelz, Matthias Dorfer, Filip Korzeniowski, Sebastian Böck, Andreas Arzt, and Gerhard Widmer. On the potential of simple framewise approaches to piano transcription. In Michael I. Mandel, Johanna Devaney, Douglas Turnbull, and George Tzanetakis, editors, *Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016, New York City, United States, August 7-11, 2016*, pages 475–481, 2016.

[Koepke *et al.*, 2020] A. Sophia Koepke, Olivia Wiles, Yael Moses, and Andrew Zisserman. Sight to sound: An end-to-end approach for visual piano transcription. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1838–1842, 2020.

[Kong *et al.*, 2022] Qiuqiang Kong, Bochen Li, Jitong Chen, and Yuxuan Wang. Giantmidi-piano: A large-scale midi dataset for classical piano music. *Transactions of the International Society for Music Information Retrieval*, May 2022.

[Lin *et al.*, 2022] Kevin Lin, Linjie Li, Chung-Ching Lin, Faisal Ahmed, Zhe Gan, Zicheng Liu, Yumao Lu, and Lijuan Wang. Swinbert: End-to-end transformers with sparse attention for video captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 17928–17937. IEEE, 2022.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9992–10002. IEEE, 2021.

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101, 2017.

[Marták *et al.*, 2024] Lukáš Samuel Marták, Patricia Hu, and Gerhard Widmer. Quantifying the corpus bias problem in automatic music transcription systems. In *1st International Workshop on Sound Signal Processing Applications (IWSSPA) 2024*, 2024.

[Raffel *et al.*, 2014] Colin Raffel, Brian McFee, Eric J. Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, and Daniel P. W. Ellis. Mir_eval: A transparent implementation of common MIR metrics. In Hsin-Min Wang, Yi-Hsuan Yang, and Jin Ha Lee, editors, *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*, pages 367–372, 2014.

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei.

Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

[Schedl *et al.*, 2014] Markus Schedl, Emilia Gómez, and Julián Urbano. Music information retrieval: Recent developments and applications. *Found. Trends Inf. Retr.*, 8(23):127261, sep 2014.

[Su *et al.*, 2020] Kun Su, Xiulong Liu, and Eli Shlizerman. Audeo: Audio generation for a silent performance video. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[Suteparuk, 2014] Potcharapol Suteparuk. Detection of piano keys pressed in video. *Dept. of Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep*, 2014.

[Tong *et al.*, 2022] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[Wan *et al.*, 2015] Yulong Wan, Xianliang Wang, Ruohua Zhou, and Yonghong Yan. Automatic piano music transcription using audio-visual features. *Chinese Journal of Electronics*, 24(3):596–603, 2015.

[Wang *et al.*, 2021] Xianke Wang, Wei Xu, Juanting Liu, Weiming Yang, and Wenqing Cheng. An audio-visual fusion piano transcription approach based on strategy. In *2021 24th International Conference on Digital Audio Effects (DAFx)*, pages 308–315, 2021.

[Wang *et al.*, 2023] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae V2: scaling video masked autoencoders with dual masking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 14549–14560. IEEE, 2023.

[Ycart *et al.*, 2020] Adrien Ycart, Lele Liu, Emmanouil Benetos, and Marcus T. Pearce. Investigating the perceptual validity of evaluation metrics for automatic piano music transcription. *Transactions of the International Society for Music Information Retrieval*, Jun 2020.

[Zhang *et al.*, 2022] Huan Zhang, Jingjing Tang, Syed Rifat Mahmud Rafee, and Simon Dixon György Fazekas. Atepp: A dataset of automatically transcribed expressive piano performance. In *ISMIR 2022 Hybrid Conference*, 2022.