Spatio-Temporal Analysis of Patient-Derived Organoid Videos Using Deep Learning for the Prediction of Drug Efficacy

Leo Fillioux[1], Emilie Gontran[2], Jérôme Cartry[2], Jacques RR Mathieu[2], Sabrina Bedja[2], Alice Boilève[2], Paul-Henry Cournède[1], Fanny Jaulin[2], Stergios Christodoulidis[1], Maria Vakalopoulou[1]
[1]CentraleSupélec, Université Paris Saclay first.last@centralesupelec.fr
[2]Gustave Roussy first.last@gustaveroussy.fr

Abstract

Over the last ten years, Patient-Derived Organoids (PDOs) emerged as the most reliable technology to generate ex-vivo tumor avatars. PDOs retain the main characteristics of their original tumor, making them a system of choice for pre-clinical and clinical studies. In particular, PDOs are attracting interest in the field of Functional Precision Medicine (FPM), which is based upon an ex-vivo drug test in which living tumor cells (such as PDOs) from a specific patient are exposed to a panel of anti-cancer drugs. Currently, the Adenosine Triphosphate (ATP) based cell viability assay is the gold standard test to assess the sensitivity of PDOs to drugs. The readout is measured at the end of the assay from a global PDO population and therefore does not capture single PDO responses and does not provide time resolution of drug effect. To this end, in this study, we explore for the first time the use of powerful large foundation models for the automatic processing of PDO data. In particular, we propose a novel imaging-based high-throughput screening method to assess real-time drug efficacy from a time-lapse microscopy video of PDOs. The recently proposed SAM algorithm for segmentation and DINOv2 model are adapted in a comprehensive pipeline for processing PDO microscopy frames. Moreover, an attention mechanism is proposed for fusing temporal and spatial features in a multiple instance learning setting to predict ATP. We report better results than other non-time-resolved methods, indicating that the temporality of data is an important factor for the prediction of ATP. Extensive ablations shed light on optimizing the experimental setting and automating the prediction both in real-time and for forecasting.

1. Introduction

Precision medicine aims to optimize the choice of drug given the characteristics of the patient, so as to optimize certain aspects such as the efficacy of the treatment or quality of life of the patient. Although doing this on a case-by-case basis by a clinician seems impractical, artificial intelligence-driven tools help guide this approach. In this objective, FPM [23] bases this optimization on tests performed on live patient cells.

Patient-derived organoids (PDOs) have gained great interest over the last few years as they represent minimalistic models to mimic essential features from the tissue they originate from. In the context of cancer therapy, drug efficiency can be limited due to the development of resistance in patients as well as other evolutionary changes in the tumors over time. PDOs represent a good testbed for physicians, researchers, and patients to assess personalized drug efficacy, based on tumor-specific patient characteristics.

The gold standard method to assess drug efficacy on cells relies on Adenosine Triphosphate (ATP), an energy molecule released in active cells through metabolic reactions. ATP is a biomarker for evaluating the number of viable cells. The quantity of released ATP measured by luminescence is proportional to the number of living cells. Thus, ATP quantity assessed by luminescence counts serves as a readout to evaluate drug efficacy with the estimation of remaining living cells in the experimental sample. This test is easily implementable in bench labs and reliable for assessing the global cell population response to drug exposure [37]. However, since it causes cell lysis, the ATP test is a destructive assay, that does not allow to assess real-time organoid drug response, and post-ATP observations to evaluate long-term viability changes and drug resistance.

The emergence of large foundation models in various fields of machine learning has allowed for novel solutions in a variety of downstream tasks. Notably, SAM (Segment Anything Model) [22] or SEEM (Segment Everything Everywhere All at Once) [50] have facilitated segmentation by providing a general model for segmentation from various types of prompts, which is extremely valuable for tasks for which little to no annotations can be found. Similarly, self-supervised [11, 4, 14, 9] or unsupervised models such as DINOv2 [34] provide high-quality descriptors for data

and facilitate the training for downstream tasks with relatively small data and task-specific samples. This training setting allows the extracted features to be task-agnostic and, therefore, to adapt more easily to new tasks. Building on these recent advances, in this paper, we propose a new high-throughput screening method for the analysis of PDOs testing multiple drugs and high-quality videos. Indeed, processing of this data is usually performed in a manual and time-consuming setting.

In this paper, our contributions are the following. To our knowledge, we present the first fully automatic method for processing high-quality videos of PDOs, conducting a spatio-temporal analysis for the prediction of ATP. We propose an efficient, automatic, and accurate prompt engineering paradigm taking into account the temporal characteristics of PDOs for using SAM without the need for additional training. Finally, we explore powerful recent foundation models for the spatio-temporal representation of PDOs for the first time coupling them with time sequence modeling and multiple instance learning.

An extensive experimental analysis was performed to identify the best representations for PDOs as well as the most informative time frames, which we used for the accurate prediction of ATP. This opens the possibility to integrate any other clinical endpoint, to match PDO drug sensitivity and patient clinical response.

2. Related work

Foundation Models in Computer Vision. Foundation models are precious resources and stimulate research by both providing a strong solution for a specific task and by proposing a novel innovative approach. The authors of the "Florence" model [46] introduced a method for learning joint visual-textual features that can be adapted to a multitude of joint tasks. The CLIP model [38] uses a contrastive learning approach in order to learn image embeddings, guided by an associated textual description of the image, which can transfer zero-shot to a wide variety of tasks. This approach has led to many extensions [25, 32, 24]. DINOv2 [34] is a deep learning model trained in a self-supervised manner that leverages more stable training at a bigger scale for better taking advantage of large datasets. The model is trained on a dataset of 142M images. It is based on the ViT architecture [13] and is made available in different sizes. Very recently, foundation models for image segmentation such as SegGPT [43], SEEM [50], and SAM [22] have surfaced. SAM provides a strong zero-shot transfer approach by combining a powerful image encoder with a prompt encoder which can adapt to multimodal prompts. Its convincing adaptability led to many adaptations for videos [45], lightweight versions [49], or medical applications [44, 26]. While these models show impressive results and generalizability, their adaptations to medical applications often show limitations. For example, recent generative models such as GLIDE [33] or DALL-E [39] show impressive results on natural images but struggle to generate realistic medical images without prior fine-tuning [21, 2].

Time Sequence Modeling. Modeling the temporal aspect of data can be achieved either on extracted features of the frames or directly in an end-to-end fashion on the videos. Once features have been extracted from frames of the videos, standard sequence models such as LSTMs [18] or Transformers [42, 47] can be applied for a variety of tasks. They aim to capture dependencies between elements of a sequence, either in the form of recurrent neural networks or self-attention mechanisms. State space modeling is gaining attraction in sequence modeling, especially for long sequences [15], and more specifically for modeling time series [48]. For end-to-end video modeling, Transformer adaptations are very popular such as the Video Swin Transformer [27], ViViT [3], or the TimeSformer [5]. We are also beginning to see adaptations of large language models (LLMs) for video understanding tasks [10]. However, even if these models are available, their application to PDOs has not yet been explored and investigated, especially in a low-availability data regime.

Analysis of PDO Microscopy Images. With the emergence of PDOs as a key technology for FPM in the last few years, machine learning methods to analyze this data have followed. Earlier studies [6, 30] looked for methods for tracking and analyzing the dynamics of PDOs through time. Authors of "D-CryptO" [1] proposed to classify types of organoids based on their morphology. Later research sought to use organoid analysis pipelines for other downstream tasks, such as the prediction of kidney differentiation [35] or the prediction of a biomarker of Huntington's disease [31]. Recently [7] have developed a pipeline for predicting ATP from microscopy images at single time points in a multiple instance learning setting. However, all these methods focus on analyzing single representations of organoids, losing a lot of information that arises from their temporal dynamics for different drug treatments. Moreover, they are mainly based on classical processing techniques, such as the extraction of predefined visual features, the extraction of ResNet features, or segmentation through biological staining.

3. Methods

In this section, we will start by describing the specificities of the dataset and the preprocessing steps, and then proceed to an in-depth presentation of the proposed method. Specifically, we will present the methodology for the segmentation of the cavities and organoids by employing SAM [22], then we will present the details of the feature extraction from the organoid regions using DINOv2 [34]. The model for the final prediction of the ATP given the features from the cavities under the multiple instance learning

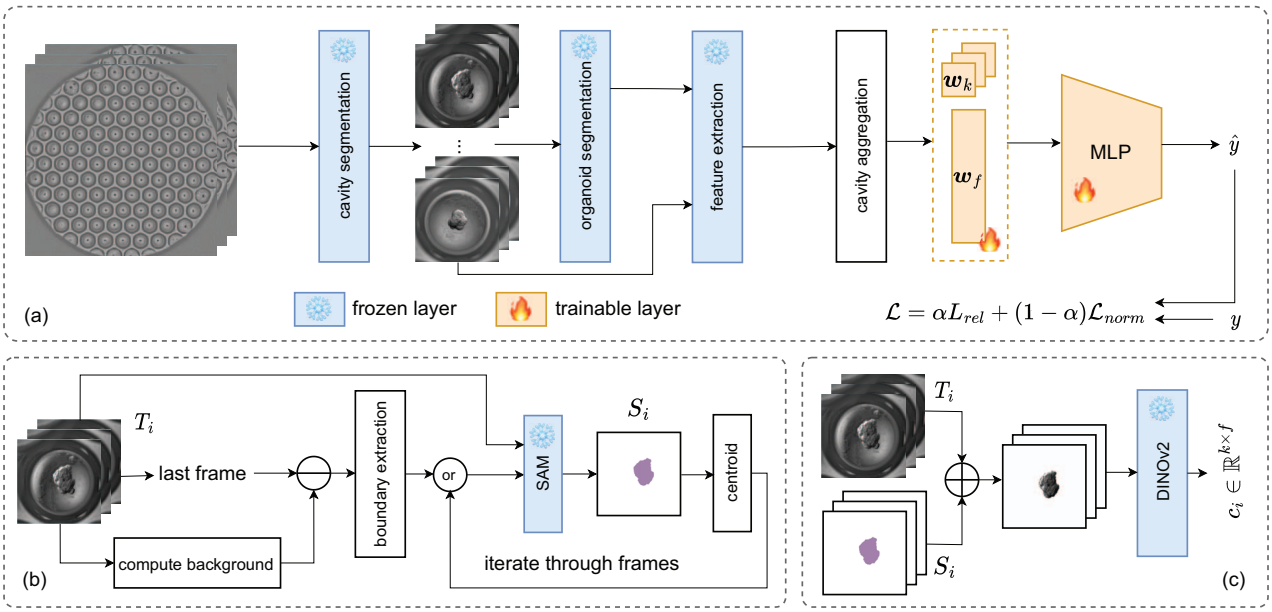


Figure 1. Illustration of our proposed pipeline. (a) The overall pipeline taking as input the whole well timelapse and outputting the predicted ATP $\hat{y}$. (b) Organoid segmentation based on automatic generation of prompts for SAM [22]. (c) Feature extraction given input frames and associated segmentation maps focusing in the region of interest and using DINOv2 model [34].

framework will then be presented. Figure 1 illustrates the proposed pipeline.

3.1. Data

To capture real-time information about drug efficacy on patient organoids, bright-field imaging techniques represent a good non-invasive alternative to the ATP bioluminescence test. However, having single-organoid resolution may be a challenging task when working with the standard experimental setup where organoids are embedded in hydrogel droplets, which results in heterogenous organoid distribution at different depths and subsequently heterogeneous organoid sizes, making the drug efficacy assessment less accurate. To overcome these issues, we developed a high-throughput experimental and imaging pipeline. It consists in using high-throughput cell culture systems with 96-well plates containing 500 μm diameter cavities. Each cavity contains one organoid, while the setting allows for the parallel follow-up of individual organoids with a homogeneous depth distribution. Testing multiple drugs in parallel thus becomes easier with one image containing the information of multiple organoids under the same treatment. One type of drug at a given concentration is applied to each well, and the ATP can only be measured on the well level. Figure 2 shows one well containing multiple cavities together with their automatic segmentation. A total of 116 wells are imaged, from which we detect 8241 cavities, and therefore 8241 organoids.

A cavity is imaged in a set of frames $T = \{T_1, ..., T_f\}, T_i \in \mathbb{R}^{w \times h}$, where w and h are the frame dimensions, and f the number of frames in each timelapse. In this study, we developed a novel pipeline to segment each cavity $S = \{S_1, ..., S_f\}, S_i \in \mathbb{R}^{w \times h}$ and using S and T, we extract interesting features from each organoid. Finally, we model a well by a set of cavity features $\mathcal{C} = \{c_1, \ldots, c_n\} \in \mathbb{R}^{n \times k \times f}$, where n is the number of cavities in the well, and k the dimensionality of the feature space. The final ATP value is defined as $y \in \mathbb{R}$.

The patient-derived organoids (from now on called organoids, for simplicity) are extracted from colorectal cancer patients. In this analysis, we do not consider the concentration of the drugs and only observe their effects indirectly with the ATP. Drugs are introduced one day after the organoids have been formed in the cavities. The wells are imaged every 30 minutes for 100 hours, resulting in $f = 200$ frames for each well. Wells are digitized using an Agilent Lionheart FX digital microscope. Note that some experiments could take slightly longer than 200 frames, for those cases, only the last 200 frames of the experiments were retained, in order to keep the measurement of the ATP at the last frame.

3.2. Preprocessing.

Each scan consists of acquiring quarters of wells at three different depth levels, resulting in 12 images for each time point. A z-projection based on Sobel filters [20] is used to project the images into the same plane, before stitching all four corners into a single image. Artifacts appear

through time (e.g. evaporation of the liquid impacting the contrast), which may impact the segmentation pipeline. To account for this, we normalize the contrast of the timelapse through time. Finally, all frames of a timelapse are then co-registered using SIFT-based features [29]. All preprocessing is performed using ImageJ [40]. Figures 2a-2d show all steps of the preprocessing from the 12 input images per frame to the normalized frame.

3.3. Automatic Segmentation of Regions of Interest

In this study, we propose and develop an automatic pipeline based on SAM [22] for the automatic segmentation of regions of interest in the video of the organoids. For accurate segmentation, various characteristics of these regions are used and different prompt engineering strategies have been developed to handle the nature of the data and their temporal information.

Cavity Segmentation. The objective is to segment the individual cavities contained in the well, in order to represent the well as a set of cavities. We assume that the detection of the cavities only needs to be run in a single frame, as the frames of the video have been co-registered. The well is considered as the mask with the biggest surface area containing the center pixel in the frame is considered, as it is the biggest object in the frame. Given the candidate masks $\mathcal{M} = \{m_1, \ldots, m_n\}$, $m_{\text{well}} = m_{argmax_i\{\text{area}(m_i)\}}$. The cavities are defined as other regions detected by SAM, which have a circularity above a certain threshold and which are included in the detected well. For each point b_i on the boundary B of a detected region m, we define $\mathcal{P} = \{\max_i(\text{dist}(b_i, B))\}$, which represents for each point on the contour, the distance to the point on the contour that is furthest away. The ratio $\frac{\min \mathcal{P}}{\max \mathcal{P}}$ defines the circularity. On a perfect circle, all values in $\mathcal{P}$ should be equal to the circle's diameter and therefore have circularity $= 1$. Once the cavities have been detected on a single frame, a crop around the cavity is extracted for all frames, giving a few dozen cavity timelapses per well. Figure 2e shows an example of the cavity segmentation masks on a normalized frame.

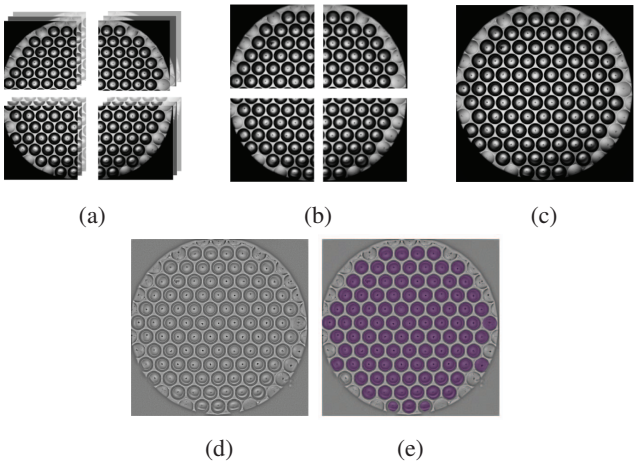


Figure 2. Example of the steps needed for preprocessing and cavity segmentation. (a) All images needed to construct one frame (4 corners at 3 different z-levels each). (b) Projected corners. (c) Stitched frame. (d) Local contrast normalization applied to the stitched frame. (e) Cavity segmentation masks.

Organoid Segmentation. Segmentation of organoids is performed on the extracted videos of the cavities by designing the proper prompts for SAM [22]. The first prompt is generated for the last frame of the timelapse T_f (as it is the frame where the organoid is the largest, and it is, therefore, more likely to find a point inside the region of interest). Canny edge detection [8] is used to extract rough contours from the last frame, after having subtracted the average frame over time, in order to remove the background. These rough contours are then filtered using a series of binary morphological operators, before extracting a centroid, which is used as a positive point prompt for SAM. Figure 3 shows an example of these steps for the last frame. Once the mask for the last frame has been generated, the prompts will be generated backward from the last frame to the first frame. The center of the organoid mask is extracted from the prediction and an exponentially weighted average from the prompts from frames T_{t+1} to T_{t+10} is used to generate the prompt for frame T_t. Multimask output is used to generate the top 3 masks, and the mask which has the highest Dice score with the generated mask from the previous frame (T_{t+1}) is chosen. Postprocessing is performed on the mask by only selecting the detected region with the highest surface area. Algorithm 1 represents the whole organoid segmentation pipeline.

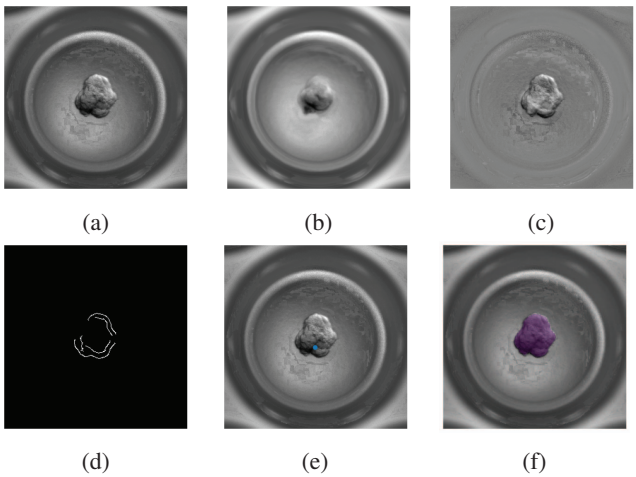


Figure 3. Example of the steps used in the generation of the prompt for the segmentation of the last frame of a cavity timelapse. (a) Last frame of the timelapse. (b) Mean frame across time. (c) Difference between the last frame and the mean frame. (d) Result of Canny edge detection on (c). (e) Generated prompt for the frame. (f) Predicted mask.

Algorithm 1 Organoid segmentation.

Inputs:
frames $T = \{T_1, ..., T_f\}, T_i \in \mathbb{R}^{w \times h}$
Outputs:
segmentations $S = \{S_1, ..., S_f\}, S_i \in \mathbb{R}^{w \times h}$
estimated_boundary= $\mathit{Canny}(T_f - \mathit{mean}(T))$
prompt_f = *centroid*(estimated_boundary)
$S_f = \mathit{SAM}(\text{prompt\_f}, T_f)$
all_prompts = {prompt_f}
for i=(f-1); i=1; i - - **do**
prompt_i = *exp_weighted_average*(all_prompts, 10)
$S_i = \mathit{SAM}(\text{prompt\_i}, T_i)$
all_prompts.insert(prompt_i)
end for

3.4. Feature Extraction

Features are extracted from each frame of the cavity using the DINOv2 [34] model, which produces task-agnostic visual features from images. A lighter version (86M parameters) of the model is used, which produces k-dimensional feature vectors. Features are extracted per frame, using a crop of the cavity around the organoid, and by masking everything outside of the regions of interest, giving $c_i = \text{DINOv2}(T_i \odot S_i)$. The model generates feature vectors of dimension $k = 768$ for each time frame.

3.5. ATP Prediction

Having access to features for each individual cavity but only having the ATP measure on the well level, the multiple instance learning setting seems to be the best approach to the problem. Each set of cavities $\mathcal{C}$ is associated with an ATP measure y. The ATP prediction model $\mathcal{M}$ should map an input $\mathcal{C} \in \mathbb{R}^{n \times k \times f}$ to an output $\hat{y} \in \mathbb{R}$. The aggregation of the cavity representations on the well level is done by mean pooling in order to obtain a well-level representation $\mathit{mean}(\mathcal{C}) \in \mathbb{R}^{k \times f}$.

Each time frame may have a different impact on the prediction of the ATP. In fact, we expect later time frames to have more impact than earlier time frames. We give the model the freedom to learn how much weight to put to each time frame using a normalized weight vector $\boldsymbol{w}_t \in \mathbb{R}^f$ that is used in a weighted average across time frames. This ensures that all cavities across all wells share the same importance for a given time frame. A feature-wise weight vector $\boldsymbol{w}_k \in \mathbb{R}^k$ is used in order to learn the relative importance of each feature that is shared across all wells.

The last element of the model is a multilayer perceptron (MLP) which is simply composed of four linear layers, with PReLU [17] activation function.

$$\begin{aligned} \mathcal{M}\colon \mathbb{R}^{n \times k \times f} &\to \mathbb{R} \\ \mathcal{C} &\mapsto \mathit{MLP}(\mathit{mean}(\mathcal{C}) \cdot \frac{\boldsymbol{w}_t}{\|\boldsymbol{w}_t\|} \odot \boldsymbol{w}_k) \end{aligned} \tag{1}$$

The loss function used for training is composed of two elements, a relative L_1 loss, to directly optimize for the mean absolute percentage error (MAPE), and a L_1 loss, which has been normalized with the maximum ATP value in the training set to have a comparable range to the relative L_1 loss.

$$\mathcal{L}_{\mathit{rel}}(y, \hat{y}) = \frac{|y - \hat{y}|}{y} \tag{2}$$

$$\mathcal{L}_{\mathit{norm}}(y, \hat{y}, y_{\mathit{max}}) = \frac{|y - \hat{y}|}{y_{\mathit{max}}} \tag{3}$$

$$\mathcal{L}(y, \hat{y}, y_{\mathit{max}}) = \alpha \mathcal{L}_{\mathit{rel}} + (1 - \alpha)\mathcal{L}_{\mathit{norm}} \tag{4}$$

Where y is the ATP ground truth, $\hat{y}$ the output of our model, y_{max} the maximum value of ATP in the training set, and α the weight given to $\mathcal{L}_{\mathit{rel}}$. A relative L_1 loss is used because of the varying range that the ATP can take within experiments. The normalized L_1 loss is used to ensure that the model still has an acceptable performance on higher values of ATPs.

3.6. Implementation Details

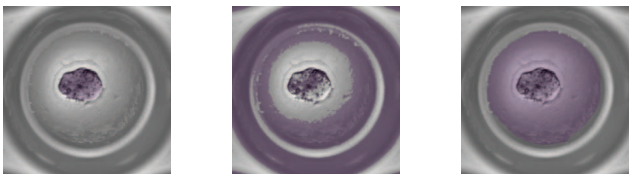
Over all experiments, training was performed under the same scheme. Cross-validation is performed over 4 splits, on the well level, resulting in training sets of size 87 and validation sets of size 29. The reported results are the average over all folds of the validation sets. All trainings were performed using the PyTorch deep learning library [36] in Python. The AdamW [28] optimizer was used with a learning rate of $1 \cdot 10^{-3}$, weight decay of $1 \cdot 10^{-1}$, and a batch size of 8. The model was trained over 2000 epochs, and early stopping with a patience of 200 epochs was used. The α parameter which balances both terms of the loss was set at $\alpha = 0.5$ after grid search. All trainings were accelerated using Nvidia Tesla V100 GPUs.

4. Experiments

To evaluate the performance of the method we used the MAPE and Pearson correlation coefficient metrics. All performances are measured on each validation set, utilizing all four models trained on the corresponding training sets. The MAPE, measures the mean relative error between the prediction $\hat{y}$ and the label y. Because of the very wide range of values of the ATP (both within an experiment and between different experiments) as presented in Table 1, the MAPE appears suitable for ensuring comparability, in particular for

ATP bin (10^5)	$[1.1, 2.8)$	$[2.8, 4.5)$	$[4.5, 6.2)$	$[6.2, 8.0)$	$[8.0, 9.7)$	$[9.7, 11.4)$	$[11.4, 13.1)$	$[13.1, 14.9]$
Count	23	19	11	15	0	7	17	24
MAPE ↓	0.34	0.15	0.08	0.07	NA	0.10	0.23	0.12

Table 1. Distribution of ATP values in our dataset, with associated MAPE.



(a) (b) (c)

Figure 4. Comparison of three methods of segmentation. (a) Our proposed method. (b) Otsu thresholding based segmentation. (c) Using SAM [22] without any prompts and choosing the mask with the highest predicted IOU.

future studies which may use datasets with different distributions of ATP. Lower values of MAPE indicate better performance.

$$MAPE(y, \hat{y}) = \frac{|y - \hat{y}|}{y} \tag{5}$$

The Pearson correlation coefficient quantifies the linear correlation between the set of predictions $\hat{Y}$ and the set of labels Y. Higher values of the Pearson correlation coefficient indicate better performance.

$$Pearson(Y, \hat{Y}) = \frac{\sum_i (Y_i - \bar{Y})(\hat{Y}_i - \bar{\hat{Y}})}{\sqrt{\sum_i (Y_i - \bar{Y})^2 (\hat{Y}_i - \bar{\hat{Y}})^2}} \tag{6}$$

Figure 4 qualitatively compares three methods of organoid segmentation to justify the superiority of our proposed method over simple intensity-based methods (Otsu thresholding) and justifies the use of our prompt generation for the use of SAM [22].

4.1. Results

With our proposed high-throughput screening method, we obtain an average MAPE and Pearson correlation coefficient on the four validation sets of 0.1755 and 0.9214, respectively. The relative error of 17.55% is to be put in context with the wide range of ATP values in our dataset, with a ratio between the highest and lowest values of ATP of 14. Table 1 shows the distribution of ATP values along with the corresponding performance of the model in terms of MAPE in each bin. The model performs worst on the bin with the smallest values of ATP, which may be explained by the precision of the measurement of ATP.

Feature Extraction. In Table 2, we explore the impact of four different feature extraction methods on the final results. Different classical and deep learning features have been explored to highlight the best representations for organoids. The pyradiomics Python package [41] allows the extraction of imaging features from regions of interest. For this study, we extracted features related to first-order statistics, gray-level matrices, and shape features. A total of 93 features have been extracted per frame. ResNet50 [16] (pre-trained on ImageNet [12]) is a popular classification architecture, which can be used for feature extraction when removing the classification head. It produces 2048-dimensional features. VICReg [4] is a feature extraction model trained in a self-supervised manner. For our study, we used a model based on the ResNet50 architecture, which produces 4096-dimensional features. Features extracted using models trained in a self-supervised manner (VICReg and DINOv2) seem to adapt better to different downstream tasks, compared to models trained for other specific tasks (ResNet50 on ImageNet), or classical predefined features. The superiority of our proposed model is highlighted both in terms of MAPE and correlation coefficient.

Attention Mechanism. We used two types of attention mechanisms, one attending to the temporal aspect of the data, and the other attending to the individual features. In Table 3 we perform an ablation study for three variations for each type of attention mechanism. For the temporal attention, we explored the impact of only using the last frame, multihead attention [42] (MHA), or the learnable weight vector $\boldsymbol{w}_t$. Similarly, for the feature-wise attention, we explored the impact of having no such attention, using multihead attention (MHA) or the learnable weight vector $\boldsymbol{w}_k$. In terms of temporal attention, using $\boldsymbol{w}_t$ clearly outperforms other methods, especially the multihead attention, while using the last frame still gives reasonable results. $\boldsymbol{w}_t$ seems to grasp the temporal importance of each frame. Concerning the feature-wise attention, the use of multihead attention clearly does not seem adequate for this application, while the effect of incorporating $\boldsymbol{w}_k$ compared to its absence is not readily apparent. The analysis of its weights could however be used for finer explainability of the model.

Features	Classical	ResNet50 [16]	VICReg [4]	DINOv2 [34]
MAPE ↓	0.3173	0.2174	<u>0.1847</u>	**0.1755**
Pearson ↑	0.7883	<u>0.8960</u>	0.8939	**0.9214**

Table 2. Comparison of MAPE and Pearson correlation coefficient for four methods of feature extraction. Best results are indicated in **bold**, and second best results are <u>underlined</u>.

Temporal attention Feature attention	Last frame None	Last frame MHA	Last frame $\boldsymbol{w}_k$	MHA None	MHA MHA	MHA $\boldsymbol{w}_k$	$\boldsymbol{w}_t$ None	$\boldsymbol{w}_t$ MHA	$\boldsymbol{w}_t$ $\boldsymbol{w}_k$
MAPE ↓	0.2027	0.2417	0.2070	0.2403	0.3719	0.2382	**0.1674**	0.2320	<u>0.1755</u>
Pearson ↑	0.9169	0.8358	0.9123	0.8755	0.6834	0.8586	<u>0.9209</u>	0.8327	**0.9214**

Table 3. Comparison of MAPE and Pearson correlation coefficient for three types of temporal attention and feature-wise attention. Best results are indicated in **bold**, and second best results are <u>underlined</u>.

Cavity aggregation	Min	SE [19]	Max	Sum	Mean
MAPE ↓	0.2425	0.2138	0.2092	<u>0.1797</u>	**0.1755**
Pearson ↑	0.8912	0.8922	0.9185	<u>0.9204</u>	**0.9214**

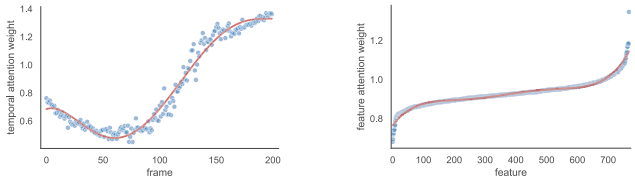
Table 4. Comparison of MAPE and Pearson correlation coefficient for four methods of feature extraction. Best results are indicated in **bold**, and second best results are <u>underlined</u>.

Cavity Aggregation. In multiple instance learning, the choice of the bag level aggregation plays a crucial role. In Table 4 we tested multiple methods of aggregation, min-pooling, max-pooling, mean, sum (which is different from the mean as the number of cavities per bag varies), and using a SE [19] block followed by a sum aggregator. The sum and mean operators outperform other types of aggregation, which can be explained by the fact that the ATP is a direct function of the number of organoid cells in the well. This information is not captured by the min-pooling and max-pooling operators. The SE block learns the weight to give to each cavity based on its features, which is used to model the fact that different cavities contribute differently to the ATP. While this may be true, we hypothesize that the information is already present in the features, and the use of the SE block therefore only adds noise.

📢 **Takeaway.** Our model has a mean MAPE 0.1755 and Pearson of 0.9214 on the validation set. Trying out different feature extraction methods shows that features from DINOv2 [34] provide the best results. Comparing the impact of a variety of feature and temporal attention schemes justifies the use of $\boldsymbol{w}_t$ and $\boldsymbol{w}_k$, while we also show the superiority of using the mean as the bag-level aggregation function.

4.2. Learned Attention Weights

Once the model is trained, analyzing the weights learned by the $\boldsymbol{w}_t$ and $\boldsymbol{w}_k$ vectors gives insights into our experiments. Figure 5a shows the weight associated with each frame in the $\boldsymbol{w}_t$ vector after training. Intuitively, we expect later frames to have the most importance. Without any constraints during training, we see that the learned temporal attention has learned a continuous and smooth distribution, meaning that nearby frames will have relatively similar importance. It is noteworthy that the initial frames seem to have more importance for the final prediction than frames after 24 hours of the experiment. This could be attributed to the fact that the organoid's initial state is significantly determinant of the final ATP, likely due to its size, as larger organoids tend to exhibit higher values of ATP.



(a) (b)

Figure 5. Visualization of the temporal attention (a) and (sorted) feature attention (b) weights for a model after training (visualization for one fold only). Polynomial fit in red.

Examining the feature importance in Figure 5b shows that the feature attention does not highlight any specific feature, as indicated by a ratio of 2 between the feature with the highest and lowest attention weight. Note that in Figure 5b, the features are sorted by their attention weight.

📢 **Takeaway.** The temporal attention vector $\boldsymbol{w}_t$ learns meaningful relative importance of the time frames, while the feature-wise attention vector $\boldsymbol{w}_k$ does not make any particular feature stand out.

4.3. Comparison to SOTA

To the best of our knowledge, only Ins-ATP [7] have explored the use of machine learning for the prediction of ATP from organoid microscopy images. They imaged multiple organoids placed in a single matrigel drop (compared to our method which has one organoid per cavity) at a single time point, which yields very different visual results to our images. We implemented the Ins-ATP method and adapted it to fit our data, by defining the bags of instances as a set of cavities from the wells, only using the last frame. This can be thought of as a compromise between the presented MeshIns-ATP and DeepIns-ATP [7] as the instances composing the bags are not learned by the model, but are chosen as meaningful regions of interest (the cavities). With this method, we obtained an average MAPE and Pearson correlation coefficient of the 4 validation sets of 0.4375 and 0.6794 respectively (compared to 0.1755 and 0.9214 respectively for our pipeline). This highlights the advances of our study in the development of a high-throughput screening method to assess real-time drug efficacy from a time-lapse microscopy video of PDOs.

📢 **Takeaway.** The Ins-ATP [7] method applied to our dataset gives lower performance than our proposed method.

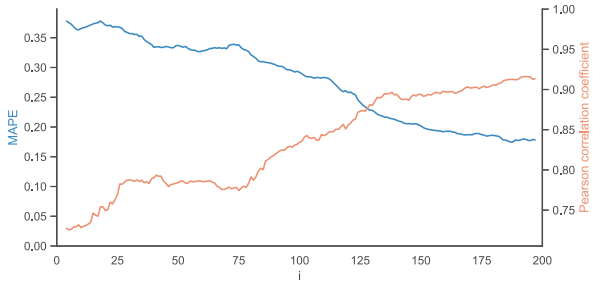
4.4. Beyond Predicting Current ATP.

We have shown the ability of our proposed pipeline to, given a sequence of frames, predict the ATP measured at the last frame. Two main questions arise. How well does our model perform in forecasting the ATP? How many frames of history does our model need to predict the current ATP? Note that while we are retraining the model with features from a subset of frames, the segmentation maps are still the ones computed on all f frames.

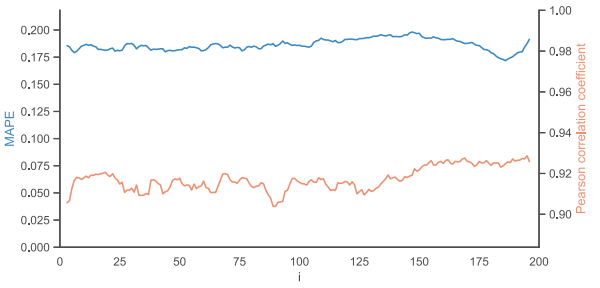
Forecasting the ATP. Given our current data (i.e. a sequence of $f = 200$ frames, and a measure of ATP at the last frame), we can evaluate the performance of our pipeline for forecasting the ATP by training the model while omitting the last frames. In Figure 6a, we show the performance of the model when trained only on frames from 0 to i, equivalent to predicting the ATP $(f - i)$ frames in advance. As expected, the performance of the model drops as the model is trained to predict ATPs earlier in the video. However, predicting 2 days in advance a MAPE of 0.28 seems reasonable compared to the best performance of 0.1755, especially when considering the wide range of ATP values.

Required history for ATP prediction. Similarly, if we omit the first frames from the training, we can evaluate how many frames of history our model needs for predicting the ATP. In Figure 6b we show the performance of the model when trained only on frames from i to f (i.e. having access to $f - i$ frames). The drop in performance is not as clear as the one shown in Figure 6a. This indicates that our model seems to perform well when not taking too many frames into account: the peak performance appears to happen when about 15 frames are given to the model. This is positive because it suggests that our model could be used for the live measurement of ATP as the images are still being acquired, as it does not need a full history of the organoid cells. However, it is important to note that although the ATP prediction part of the model seems to perform well with only 15 frames, the segmentation pipeline (and more precisely, the generation of the first prompt for SAM) relies on movement within the timelapse. We tested how well the segmentation of the organoids worked on the last 15 frames by computing the Dice score with the last 15 frames of the segmentation map computed using all frames as ground truth. The mean Dice score is 0.80, with a very uneven distribution among cavities: 75% of the cavities obtain a Dice over 0.90, while 15% obtain a Dice under 0.20. It seems that most segmentations are not affected by the use of a smaller number of frames, but for those cases which are affected, the segmentations are prone to complete failure.

Takeaway. Our model can predict the ATP in advance with a reasonable drop in performance and is able to predict with only about 15 frames of history making it usable for online predictions.



(a)



(b)

Figure 6. MAPE (left) and Pearson correlation coefficient (right) as a function of the number of frames given to the model. (a) Performance of models trained only on frames from 0 to i. (b) Performance of models trained only on frames from i to 200.

5. Conclusion

The estimation of the ATP is a standard method for the estimation of drug efficacy on organoids. However, it allows the measurement at a single timepoint for the entire experiment. In this paper, we propose a method for the spatio-temporal analysis of organoid microscopy timelapse videos, for the prediction of ATP. We assess the performance of our approach for predicting the current ATP with different ablations and we report better performance than SOTA.

Future work includes the further exploration of foundation models for the analysis of organoid videos. In this study, the foundation models are used as frozen blocks. However, authors of DINOv2 [34] showed that finetuning the model encoders to a specific dataset improved the results on the dataset. One step could be to finetune the feature extraction model to improve the representations of the organoids to be better adapted to the prediction of ATP. Similarly for the SAM [22] model, we could finetune the mask decoder part of the model, or learn more efficient prompts.

Acknowledgments. This work has benefited from state financial aid, managed by the Agence Nationale de Recherche under the investment program integrated into France 2030, project reference ANR-21-RHUS-0003. Experiments have been conducted using HPC resources from the "Mésocentre" computing center.

References

[1] Lyan Abdul, Jocelyn Xu, Alexander Sotra, Abbas Chaudary, Jerry Gao, Shravanthi Rajasekar, Nicky Anvari, Hamidreza Mahyar, and Boyang Zhang. D-crypto: deep learning-based analysis of colon organoid morphology from brightfield images. *Lab Chip*, 22:4118–4128, 2022. 2

[2] Lisa C. Adams, Felix Busch, Daniel Truhn, Marcus R. Makowski, Hugo JWL. Aerts, and Keno K. Bressem. What does dall-e 2 know about radiology?, 2022. 2

[3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer, 2021. 2

[4] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022. 1, 6

[5] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021. 2

[6] Xuesheng Bian, Gang Li, Cheng Wang, Weiquan Liu, Xiuhong Lin, Zexin Chen, Mancheung Cheung, and Xiongbiao Luo. A deep learning model for detection and tracking in high-throughput images of organoid. *Computers in Biology and Medicine*, 134:104490, 2021. 2

[7] Xuesheng Bian, Cheng Wang, Shuting Chen, Weiquan Liu, Sen Xu, Jinxin Zhu, Rugang Wang, Zexin Chen, Min Huang, and Gang Li. Ins-atp: Deep estimation of atp for organoid based on high throughput microscopic images, 2023. 2, 7

[8] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 4

[9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021. 1

[10] Guo Chen, Yin-Dong Zheng, Jiahao Wang, Jilan Xu, Yifei Huang, Junting Pan, Yi Wang, Yali Wang, Yu Qiao, Tong Lu, and Limin Wang. Videollm: Modeling video sequence with large language models, 2023. 2

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 1

[12] 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. Ieee, 2009. 6

[13] 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, 2021. 2

[14] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, and Piotr Bojanowski. Self-supervised pretraining of visual features in the wild, 2021. 1

[15] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces, 2022. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 6

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015. 5

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2

[19] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks, 2019. 7

[20] Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2):358–367, 1988. 3

[21] Jakob Nikolas Kather, Narmin Ghaffari Laleh, Sebastian Foersch, and Daniel Truhn. Medical domain knowledge in domain-agnostic generative AI. *npj Digital Medicine*, 5(1), July 2022. 2

[22] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 1, 2, 3, 4, 6, 8

[23] Anthony Letai. Functional precision cancer medicine—moving beyond pure genomics. *Nature Medicine*, 23(9):1028–1035, Sept. 2017. 1

[24] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022. 2

[25] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training, 2022. 2

[26] Yihao Liu, Jiaming Zhang, Zhangcong She, Amir Kheradmand, and Mehran Armand. Samm (segment any medical model): A 3d slicer integration to sam, 2023. 2

[27] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer, 2021. 2

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 5

[29] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004. 4

[30] Jonathan Matthews, Brooke Schuster, Sara Saheb Kashaf, Ping Liu, Mustafa Bilgic, Andrey Rzhetsky, and Savaş Tay. Organoid: a versatile deep learning platform for organoid image analysis. *bioRxiv*, 2022. 2

[31] Jakob J. Metzger, Carlota Pereda, Arjun Adhikari, Tomomi Haremaki, Szilvia Galgoczi, Eric D. Siggia, Ali H. Brivanlou, and Fred Etoc. Deep-learning analysis of micropattern-based organoids enables high-throughput drug screening of huntington's disease models. *Cell Reports Methods*, 2(9):100297, 2022. 2

[32] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training, 2021. 2

[33] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022. 2

[34] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 1, 2, 3, 5, 6, 7, 8

[35] Keonhyeok Park, Jong Young Lee, Soo Young Lee, Iljoo Jeong, Seo-Yeon Park, Jin Won Kim, Sun Ah Nam, Hyung Wook Kim, Yong Kyun Kim, and Seungchul Lee. Deep learning predicts the differentiation of kidney organoids derived from human induced pluripotent stem cells. *Kidney Research and Clinical Practice*, 42(1):75–85, Jan. 2023. 2

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 5

[37] Nhan Phan, Jenny J. Hong, Bobby Tofig, Matthew Mapua, David Elashoff, Neda A. Moatamed, Jin Huang, Sanaz Memarzadeh, Robert Damoiseaux, and Alice Soragni. A simple high-throughput approach identifies actionable drug sensitivities in patient-derived tumor organoids. *Communications Biology*, 2(1), Feb. 2019. 1

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 2

[39] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021. 2

[40] Caroline A Schneider, Wayne S Rasband, and Kevin W Eliceiri. Nih image to imagej: 25 years of image analysis. *Nat Meth*, 9(7):671–675, July 2012. 4

[41] Joost J.M. van Griethuysen, Andriy Fedorov, Chintan Parmar, Ahmed Hosny, Nicole Aucoin, Vivek Narayan, Regina G.H. Beets-Tan, Jean-Christophe Fillion-Robin, Steve Pieper, and Hugo J.W.L. Aerts. Computational radiomics system to decode the radiographic phenotype. *Cancer Research*, 77(21):e104–e107, Oct. 2017. 6

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 2, 6

[43] Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. Seggpt: Segmenting everything in context, 2023. 2

[44] Junde Wu, Yu Zhang, Rao Fu, Huihui Fang, Yuanpei Liu, Zhaowei Wang, Yanwu Xu, and Yueming Jin. Medical sam adapter: Adapting segment anything model for medical image segmentation, 2023. 2

[45] Jinyu Yang, Mingqi Gao, Zhe Li, Shang Gao, Fangjing Wang, and Feng Zheng. Track anything: Segment anything meets videos, 2023. 2

[46] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, Ce Liu, Mengchen Liu, Zicheng Liu, Yumao Lu, Yu Shi, Lijuan Wang, Jianfeng Wang, Bin Xiao, Zhen Xiao, Jianwei Yang, Michael Zeng, Luowei Zhou, and Pengchuan Zhang. Florence: A new foundation model for computer vision, 2021. 2

[47] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A transformer-based framework for multivariate time series representation learning, 2020. 2

[48] Michael Zhang, Khaled K. Saab, Michael Poli, Tri Dao, Karan Goel, and Christopher Ré. Effectively modeling time series with simple discrete state spaces, 2023. 2

[49] Xu Zhao, Wenchao Ding, Yongqi An, Yinglong Du, Tao Yu, Min Li, Ming Tang, and Jinqiao Wang. Fast segment anything, 2023. 2

[50] Xueyan Zou, Jianwei Yang, Hao Zhang, Feng Li, Linjie Li, Jianfeng Gao, and Yong Jae Lee. Segment everything everywhere all at once, 2023. 1, 2