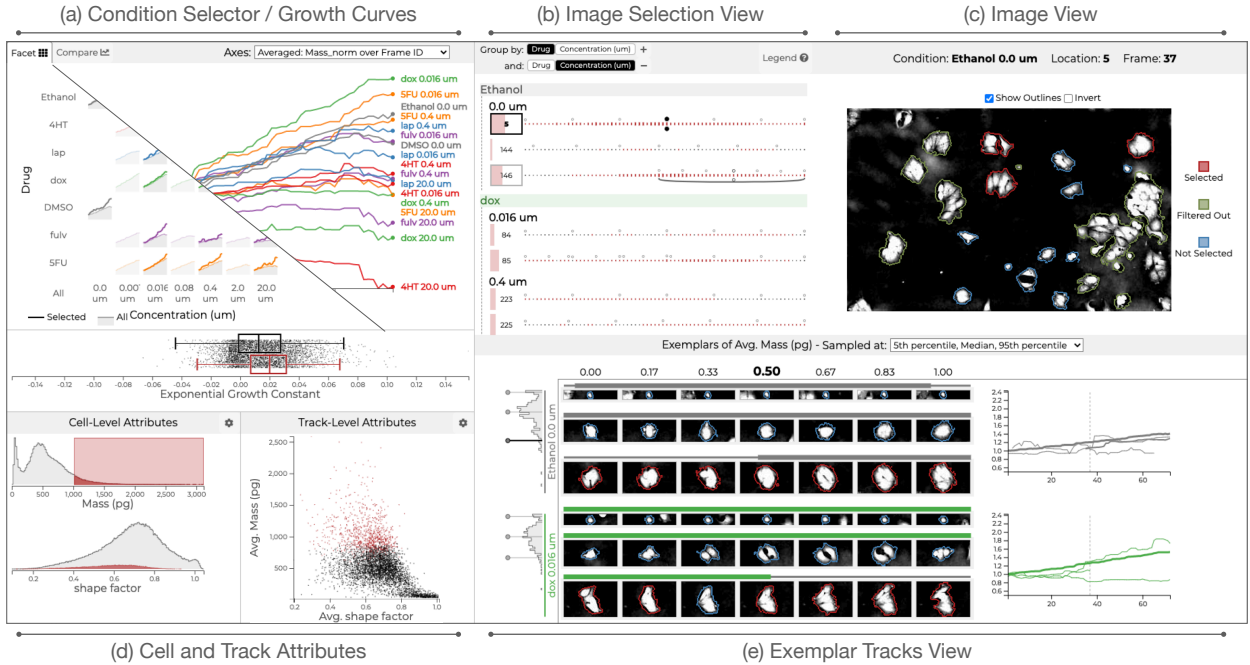# Loon: Using Exemplars to Visualize Large Scale Microscopy Data

Devin Lange, Eddie Polanco, Robert Judson-Torres, Thomas Zangle, Alexander Lex

Fig. 1: Overview of the Loon Visualization Tool. (a) The Condition Selector visualizes cell growth rates for different drugs at different concentration using small multiple line charts in a matrix. Analysts can pick conditions that show interesting behavior for detailed analysis. (b) The Image Selection View is used to navigate images and visualizes aggregate cell and track (cells tracked over time) properties. (c) The image view shows a selected microscopy image and the segmentation of cells. (d) The Cell and Track Attributes view shows distributions of or correlations between attributes of cells and tracks and serves as the primary means to define selections and filters. (e) The Exemplar Cells view shows cells extracted from the images and samples from a user-specified distribution. It also shows the growth curves for the condition and the selected cells.

Abstract—Which drug is most promising for a cancer patient? This is a question a new microscopy-based approach for measuring the mass of individual cancer cells treated with different drugs promises to answer in only a few hours. However, the analysis pipeline for extracting data from these images is still far from complete automation: human intervention is necessary for quality control for preprocessing steps such as segmentation, to adjust filters, and remove noise, and for the analysis of the result. To address this workflow, we developed Loon, a visualization tool for analyzing drug screening data based on quantitative phase microscopy imaging. Loon visualizes both, derived data such as growth rates, and imaging data. Since the images are collected automatically at a large scale, manual inspection of images and segmentations is infeasible. However, reviewing representative samples of cells is essential, both for quality control and for data analysis. We introduce a new approach of choosing and visualizing representative exemplar cells that retain a close connection to the low-level data. By tightly integrating the derived data visualization capabilities with the novel exemplar visualization and providing selection and filtering capabilities, Loon is well suited for making decisions about which drugs are suitable for a specific patient.

Index Terms—Microscopy visualization, cancer cell lines, exemplars, design study.
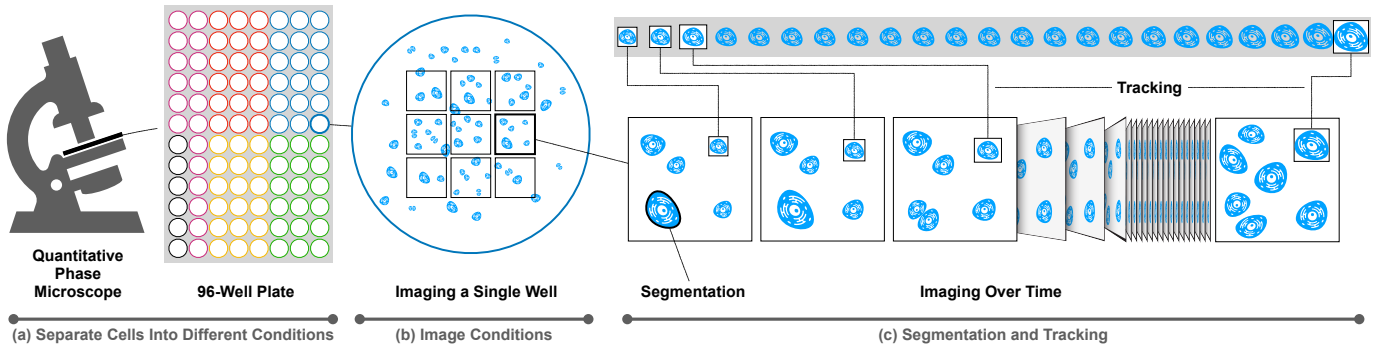
◆

1 INTRODUCTION

Automatically acquired, large-scale microscopy data is an increasingly important tool in life-science research and medical practice. Areas such as brain connectomics create high-resolution images using electron

- *Devin Lange, Eddie Polanco, Robert Judson-Torres, Thomas Zangle, and Alexander Lex are with the University of Utah. E-Mail: devin@sci.utah.edu, eddiepolanco@chemeng.utah.edu, Robert.Judson-Torres@hci.utah.edu, tzangle@chemeng.utah.edu, alex@sci.utah.edu.*

microscopes of neurons and use segmentation to reconstruct the connectivity of the brain [1, 21]. Similarly, high-throughput screening to observe the effect of drug candidates on their cell lines is a frequently used method in pharmacological research [12]. These approaches have in common that the bottleneck has moved from acquiring image data to processing and analyzing the data. With the many thousands if not millions of images captured, these pipelines heavily rely on automatic image analysis processes, such as segmentation, as the basis for deriving datasets of interest. However, completely automated setups have proven elusive, not least due to the heterogeneity of the images and biological structures captured [1, 30]. Analysts face many challenges, including having to conduct quality control of automatic processes (checking segmentation and tracking) and adapting the analysis for

Fig. 2: The data acquisition process. (a) Cancer cells are plated in wells, and each well is treated either with a control substance or one of several cancer drugs (indicated as colors) at different concentrations. (b) For each well, multiple segments are imaged. (c) Individual cells are segmented. By repeatedly imaging each location as time passes, tracks of cells can be computed. The cell segmentation and tracking information can then be used to compute attributes such as mass and growth rates.

the particular experiment. To address these challenges, analysts need support through interactive visual analysis systems.

In this paper, we introduce Loon, a visual analysis tool for a novel type of application: screening specific patient's tumor samples for a variety of cancer drugs using quantitative phase imaging (QPI) data. Our collaborators are developing this novel technology and the associated data analysis pipeline. The goal is to rapidly (i.e., within 1–2 days) experimentally determine which known cancer drug inhibits cell growth, thereby taking a leap towards personalized medicine. The heterogeneous nature of the data — for different tumor types taken from different patients — requires a flexible visual analysis solution.

Our contribution is two-fold: on the one hand, we contribute a design study, based on a detailed analysis of the domain problem, which results in a functioning and deployed software tool. On the other hand, we also contribute a novel technique to visualize representative exemplars of cells by sampling cells along user-specified data dimensions. This approach enables analysts to both quickly validate pre-processing steps such as segmentation and tracking, and analyze the properties of cells in different conditions, thereby making visual analysis of microscopy data feasible even for large datasets. We argue that this technique can be applied broadly to other imaging/segmentation problems and beyond.

We validate our design in two case studies —— one focused on quality control, the other applied to data analysis — and through examples with two different datasets.

2 RELATED WORK

Our work is most closely related to visualizing cell microscopy data. However, we also briefly discuss other approaches of visualizing representative samples in datasets.

2.1 Cell Microscopy Data Visualization

Automated microscopy approaches, and in particular live-cell imaging — where cells are observed while they develop — have entered the mainstream of biological research in the last decade. Data analysis and visualization of imaging data have been identified as critical in the life science community [15, 33], and a recent survey by Pretorius et al. [30] has recognized it as an important emergent visualization research area.

Pretorius et al. also introduce six classes of visualization methods for live-cell imaging data [30]. In Loon, we leverage a variety of them, including spatial embedding (for visualizing cells in their field of observation), temporal plots (including at the aggregated and faceted level), and aggregate visualizations of classes or features. Notably, our faceted view for comparing growth curves between conditions is related to the approach by Khan et al. [22], but also to Pathline [28] and MulteeSum [27], although the latter two tools use such a matrix in a different application context. Our visualization of exemplars, however, does not fit neatly in Pretorius et al.'s classification.

Many commercial and academic microscopy analysis tools provide segmentation and quality control methods. CellProfiler [7] is a widely used open-source image processing pipeline for microscopy data, and provides the means of refining segmentations, sharing some of our

analysis goals. In the context of connectomics, both semi-automatic segmentation approaches [19] as well as proof-reading workflows for cell segmentations [1] have been developed.

Our collaborators use both custom and commercial systems. The commercial systems they use — Livecyte [2] and HoloMonitor [31] — are packaged with analytical software. Both tools provide various statistical plots to aid in the analysis. Although the quality of the images can be easily monitored with these software packages, determining the quality of the segmentation and tracking and filtering inappropriately segmented objects is onerous. Our approach of providing cell track exemplars that recalculate in real-time as filters are applied to object features makes quality checks much more efficient, yet our tool remains compatible with the data generated from these commercial platforms.

Other tools support visualization of imaging data and derived statistical metrics. CellProfilerAnalyst [18], for example, provides such visualization interfaces. Notably, the most recent version of CellProfilerAnalyst also provides a gallery view of extracted cells [11], though in contrast to our approach, the cells are not sampled to provide a representative picture, and they don't consider a temporal dimension. Hamilton et al. [14] analyze the similarity of microscopy images and arrange them in 2D such that similar images are close to each other. They also compute and show a set of representative images. In contrast to our work, they operate on an image level, i.e., they don't extract exemplar cells or other components. Krueger et al. [23] embed individual cells in a plot using dimensionality reduction, but they don't show representative samples of certain classes.

The tool most closely related to our own is ScreenIt [12], which is designed for high-throughput screening and high-content screening for drug discovery. ScreenIt also combines image data with visualization of attributes, and also uses exemplars, though, in contrast to our work, exemplars are selected manually and then used to define a phenotype of interest. Our use of exemplars is different, as we automatically pick a suite of exemplars for a user-selected data attribute so that analysts can understand typical and outlier cases for these conditions.

2.2 Exemplars in Different Contexts

Using representative examples to visualize classes of objects is a powerful approach. Many clustering algorithms, for example, can provide a "typical" member of a cluster that can then be used to represent a class. For example, Bach et al. [4] aggregate groups of temporal networks represented as adjacency matrices into piles based on clustering and show an example to represent the pile. Lekschas et al. [24] take a similar approach for genome interaction matrices, but also project the piles and the associated examples into 2D space based on similarity. In the context of computer vision, Carter et al. have developed an activation atlas that uses a structured layout of "activation images" and projects them into 2D space [8], leveraging exemplars to aid in the understanding of how a neural network sees.

While these examples illustrate that exemplars are a powerful way to aid analysts in understanding a large dataset, they are quite distinct from our approach and from our application area.

3 Biological Background

Cancer is a complex and dynamic disease with individual tumors presenting substantial genomic and transcriptomic heterogeneity. This makes it difficult to select the appropriate therapy for treating an individual cancer patient. Functional precision medicine seeks to guide treatment decisions using assays that measure the response of patient-derived tumor cells to candidate therapies [13]. Our collaborators are demonstrating the use of quantitative phase imaging as a method to measure the growth rate of individual cancer cells in response to chemotherapy. Effective chemotherapies reduce cell growth rates at low concentrations, indicating sensitivity. This work is therefore a step towards a functional precision assay.

QPI measures the phase shift of light as it passes through and interacts with cell mass. This phase shift is proportional to cell mass [35]. QPI has previously been applied to rapidly (within 5–10 h) measure chemotherapy sensitivity [9] and changes in cell phenotype associated with metastatic dissemination with single-cell resolution [16, 36]. QPI is, therefore, an ideal method for assessing the response of cancer cells to potential therapies.

In a typical experiment, illustrated in Figure 2 a cancer cell line derived from a tumor sample is plated at low density (cells covering 20-50 percent of the cell culture surface) in a plate containing 24–96 individual wells. Each well is then treated with different drugs (or controls) at different concentrations. Then, multiple imaging locations for each well are selected. These locations are imaged repeatedly, once every 10–30 minutes, for a period of 1–5 days. We refer to an image in a time series as a **frame**.

To process the resulting phase images, our collaborators first use a rough segmentation to separate the cells from the background and average the background from 1000 images in order to obtain a common background that can be used to eliminate noise due to dust and other imperfections in the optical system. This common background image is then subtracted from every image in the experiment. After correcting the background, they then use a segmentation algorithm to segment cells and compute the mass of segmented cells by integrating the phase shift over the area of each cell and multiplying by the specific refractive increment. Cells are then tracked from frame to frame (see Figure 2) by minimizing the distance between cells in frame n relative to frame $n+1$, in a three-dimensional vector space such that the linearly independent components are distance traveled in x, distance traveled in y, and change in mass.

4 Dataset Description

The Zangle Lab and the Judson-Torres Lab are currently using Loon to analyze the datasets they collect. In this paper, we focus on one dataset from each lab as examples, though Loon is compatible with many similar datasets.

The main dataset we use to demonstrate Loon in this paper and in the accompanying video and live demo is a breast cancer dataset. Worldwide, breast cancer is the most common cancer for women. Finding the right treatment for breast cancer could help hundreds of thousands of women who are diagnosed with breast cancer in the U.S. next year. In one experiment, the Zangle Lab exposed breast cancer cells to five drugs at six different concentrations, as well as two different control groups. This experiment used a multi-well plate to image cells at 864 locations. Each location was imaged once every 20 minutes for 24 hours, resulting in a total of 62,208 images at a resolution of 1920x1200. For this experiment, they chose not to segment individual cells but focus on cell clusters instead, as overall growth can also be tracked by observing the growth of clusters, and the segmentation is simplified. In total, this dataset contain 1,132,850 cell clusters and 166,015 cell tracks.

The second example dataset we describe in a case study is provided by the Judson-Torres Lab and is designed to investigate metastatic melanoma — skin cancer that has spread to other parts of the body. A tumor can contain subpopulations of cells that are either resistant or sensitive to treatment from a particular drug. In this experiment, the Judson-Torres Lab models this heterogeneity by combining two human melanoma cell lines. One that is resistant to drugs, and one that is sensitive. With this, they can carefully control the ratios of the two

cell types. In their experiment, they used a multi-well plate to separate four groups of cell mixture ratios (100%, 80%, 20%, 0% resistant cells). For each group, they exposed it to the drug vemurafenib at five different concentrations, as well as a control group with no drug. These conditions were imaged at 28 locations every 30 minutes for 48 hours, resulting in a total of 2,716 images (at resolution 768x768), 439,699 segmented cells, and 111,151 cell tracks.

In general, these datasets contain thousands or tens of thousands of **images** (with resolutions between 400x300 to 1920x1200) which are assigned to **conditions**. Conditions are typically a combination of a type of drug and its concentration, or controls. Each image contains dozens or hundreds of **cells**. In a preprocessing step, our collaborators compute segmentation labels and derived numerical attributes, such as the area of a cell, the estimated mass of a cell, its position, attributes describing morphology ("shape factors"), etc. The position and mass of the cells are then used to track cells at the same location over time — resulting in **tracks** — for which the change in attributes over time can be measured and visualized. We supplement this data with derived data computed on the fly on both the cell (normalized mass) and tracks level (a growth rate).

5 Collaboration and Domain Goals

Over the last 18 months, we collaborated with the above-mentioned two different labs that use the same imaging methodology, but have slightly different focus areas; members of both labs are also co-authors of this paper. Our primary collaborators are researchers in chemical engineering; their objective is to develop and validate quantitative phase imaging for the purpose of cancer drug screening. Their lab develops custom hardware setups and analysis software for QPI data analysis. Our secondary collaborators are researchers in a cancer center who use commercial hardware and software to analyze their data and are primarily interested in the biomedical research questions that can be answered with this technology.

Eliciting domain goals is a key challenge in application-oriented work. Sedlmair et al. [32] discuss that "just talking to users [..] is typically not sufficient". We used two complementary methods to understand user needs: first, we used contextual design methods [17] to develop an understanding of the domain and the analysis problems of our collaborators. The first author of this paper embedded himself in the research group by attending all lab meetings for over a year, and by closely collaborating with the graduate students and the PIs. As the project progressed, the first-author regularly collaborated with the domain experts to update their analysis code and aided them in unifying their processes across experiments, in particular between the different labs. Second, to jump-start our understanding of the domain problem, we used a creative visualization opportunity workshop [20] with five analysts for half a day. As a result of this workshop, we identified two high-level needs for our collaboration. First, there was a desire to improve the quality control process of the lab. Especially the segmenting and tracking of cells, since it is a labor-intensive task shared across the lab. The next high-level goal we identified was comparing experimental conditions. That is, identifying differences in cell behavior when exposed to different drugs. While not an explicit goal, the importance of cell growth curves in their analysis also became apparent at this workshop.

The outcome of this process is the co-design of the Loon visualization tool, which we developed to answer the following domain-specific question. The overall analysis goal is to judge the effect of different drugs at different concentrations on patient-specific tumor cell lines. Together with other information, such as the clinical history of a patient and side-effects of drugs, this information is then used for making treatment decisions. However, especially given that these systems are novel technology and given the heterogeneity of experiments and cancer, quality control and data preparation are also essential domain goals.

The quality control domain goals are concerned with validating various aspects of the experiment and the preprocessing:

QC-1 Validate Segmentation: Given the heterogeneity of tumor tissue samples and the variety of experiments, no two experiments are quite alike. In addition to different cell sizes and morphologies,

our collaborators sometimes track clusters of cells, but sometimes also individual cells. All segmentation algorithms require parameterization, and human checks are necessary. While a rough validation is possible just by looking at a few examples, systematic errors that occur in edge cases are more difficult to spot with traditional methods, such as checking a sample of images. Hence, a key QC goal is to validate segmentation for a representative and diverse sample of images.

QC-2 Validate Cell Tracking: Based on segmentations of individual cells, our collaborators compute tracks of cells (or cell clusters) that follow a cell over time. This process can also lead to errors, as the tracking can be lost if, e.g., the cell moves too much. Again, different parametrizations can lead to different results, leading to the QC goal of validating cell tracking for representative and diverse samples of cells.

QC-3 Validate Derived Attributes: Derived attributes, such as mass, size, or location in the experiment, etc., can also serve to validate all aspects of the experimental pipeline. To validate derived attributes, our collaborators need to explore these attributes and check for plausibility.

In addition to the quality control domain goals, we have identified the following data analysis domain goals:

DA-1 Data cleaning: As a preprocessing step to the analysis, the data needs to be cleaned of data points that don't correspond to cells that can be tracked. Among the common artefacts our collaborators want to remove are dust specks or debris that are mistakenly identified as small cells. Also, some tracks can be short, and our collaborators want to remove these as calculating growth rates is more accurate for longer tracks. Setting a minimum track length furthermore reduces errors due to counting cells more than once in the subsequent analysis. However, these filters can't be done completely automatically but instead have to be individually specified based on visual inspection.

DA-2 Selecting Conditions: A typical experiment will analyze control conditions (no drug) and several drugs, resulting in dozens of experiments. As a first step, analysts want to analyze growth rates for all of these conditions and pick the ones that look promising in terms of inhibiting growth at low concentrations, for subsequent detailed analysis.

DA-3 Analyze Cell Growth: The most important goal of the domain experts is to analyze growth indicators of cells of the different conditions. This supports the ultimate goal to select medications that inhibit cell growth at low drug concentrations. Depending on the experiment, the experts might want to study normalized or absolute mass or area over time, while comparing between different conditions. They also might analyze examples of fast-growing or dying cells in each condition.

DA-4 Analyzing Heterogeneity: Cancer cells are known to be heterogeneous in their genetic makeup and their responsiveness to therapy. It is possible that a particular drug might be effective against one set of cells, but not against another set of cells, both of which can be found in the same tumor. To identify such cases, our collaborators need to study cells that both grow and that shrink over time with respect to their attributes and their appearance.

DA-5 Communicating: Since our collaborators are evaluating the overall method, they also want to communicate their findings in papers and presentations. This includes sharing plots of growth rates, but also sharing images of exemplar cells.

6 LOON VISUALIZATION DESIGN

We designed Loon to address the domain-specific goals discussed in the previous section. Ultimately, we can translate these goals into five visualization tasks. First, analysts need an interface to **select conditions** (Figure 1(a)) based on cell growth rates and contextual information about the conditions, which directly maps to the selecting condition requirement (**DA-2**).

Second, for quality control goals, including evaluating the quality of segmentations (**QC-1**) and tracking (**QC-2**) we need to **visualize the images and cells** (Figure 1(b), (c) and (e)). Visualizing cells is also essential for the analysis of heterogeneity (**DA-4**), and for communicating the effect a drug has on cells (**DA-5**). The challenge with this task is scale: we need methods to visualize good representative subsets of tens of thousands of cells and images.

Third, we need the ability to **visualize tracks** (Figure 1(b) and (e)), both for quality control of the tracking (**QC-2**) and to analyze cell growth (**DA-3**). Since tracks are deeply connected to the imaging data, it is natural to visualize both of them together. Images of tracks are also difficult to curate manually; hence a visualization of tracks based on cells also served the communication goal (**DA-5**).

Fourth, we need the ability to **visualize cell growth rates** (Figure 1(a)), such as mass over time, area over time, or growth rates, and compare these between conditions in service of the main analysis goal: analyze cell growth for selecting drugs (**DA-3**).

Finally, we need the ability to **visualize distributions of and correlations between derived data items** (Figure 1 (d)), to enable quality control for derived attributes (**QC-3**), and to serve as an interface to **dynamically filter and select items**, in service of the data cleaning goal (**DA-1**) and the ability to explore heterogeneity in cell lines (**DA-4**).

A deployed version of Loon can be accessed at `http://loon.sci.utah.edu/`. This includes a publicly available demo dataset, but also an option to log-in to access more datasets for our collaborators.

6.1 Selecting Conditions

Understanding the growth of cells under different conditions is a key component of our collaborators' workflow (**DA-3**). However, a typical experiment consists of many conditions that turn out to be of little value: of the dozens of conditions and controls, only a few will show responses to the drug. As data for individual conditions can't be sensibly aggregated, it is useful to first subset the conditions by how promising they are, which allows analysts to focus on the relevant data and alleviates scalability concerns (**DA-2**).

Figure 3 shows a compact overview of all possible conditions, using a small-multiple grid view with area charts, inspired by Meyer et al.'s curvemap [28].
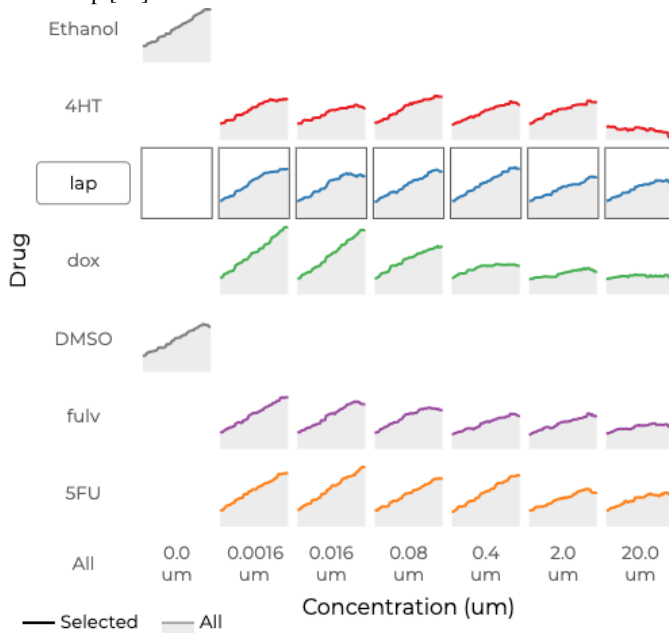
Fig. 3: Condition Selection View. Small area charts show the growth rate for individual conditions. The charts are arranged by drug or control (ethanol and DMSO) type vertically, and by concentration horizontally. Drugs are color coded; controls are shown in gray. Each condition can be selected individually, or conditions can be selected in bulk by row or column.
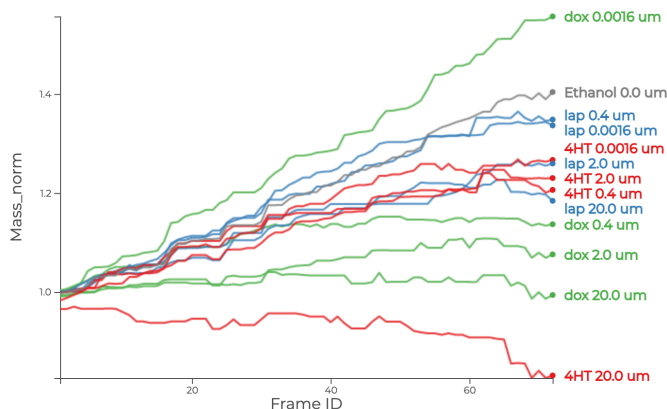
Fig. 4: Comparison view showing frame ID, a proxy for time, plotted against normalized mass for a selected subset of conditions.

Fig. 5: The image selection view enables analysts to browse images by condition (drug and concentration) and frame. It also displays statistical information about the number of cells in each frame, and visualizes track lengths for selected cells.

The area charts themselves plot cell growth over time; the charts are arranged in a matrix that plots drugs or control vs. concentrations so that charts on the right correspond to experiments with high concentration. We use color and vertical position to redundantly encode the type of drug. The colors are used consistently throughout the tool.

The area charts themselves plot cell growth over time; the charts are arranged in a matrix that plots drugs or control vs. concentrations so that charts on the right correspond to experiments with high concentration. We use color and vertical position to redundantly encode the type of drug. The colors are used consistently throughout the tool.

This view serves two purposes: to give an overview of the effectiveness of the drugs (a low or downward slope indicates inhibition) and to select conditions for detailed analysis in other views. Analysts want to select conditions that inhibit growth at a low concentration, as these drugs are likely to be effective. Conditions can be selected individually, or in bulk by row or column.

We considered several alternatives to this view, including a line-chart plotting all conditions in the same plot, as such a chart would have a larger drawing area and would make comparisons between two lines more precise. However, we found that even the 32 conditions in our sample dataset lead to significant overplotting and that the structure of drugs and concentrations is hard to understand.

6.2 Comparison View

Complementary to the selection view, we provide a comparison view that can be used to plot various metrics, such as normalized mass, mass, shape factors, area, intensity, etc., over time (or frame). These metrics can either be aggregated (averaged) by conditions or plotted for each track individually. Figure 4 shows an example of the normalized mass plotted against frame ID inline charts, a key metric to address the "analyze cell growth" goal (**DA-3**). We use the drug-specific color scheme, and show labels for every line. Compared to the selection view, this view enables a more detailed comparison between conditions, due to the larger screen area and the overlaid lines. This, however, comes at the cost of scalability and understanding the relationships between the trends and conditions.

By default, we apply a median filter to smooth out the appearance of the curves. Due to differences in focus settings, which are updated after every frame, the derived metrics, such as mass, can fluctuate, though this fluctuation is typically consistent between wells.

6.3 Image Viewer and Exemplars

A key component of Loon, and also our main technical contribution, is the ability to view and navigate microscopy images efficiently and to get a good overview of representative cells for different conditions. This is in service of the segmentation and tracking quality control goals (**QC-1**, **QC-2**) but also for the analysis of cell heterogeneity (**DA-4**). The image analysis capabilities are made up of three tightly coupled
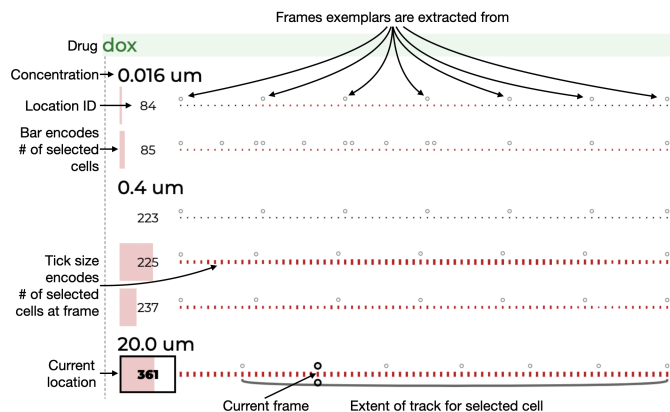
components: the image selection interface, the single image viewer, and the cell tracks display (Figure 1(b), (c), and (e)).

Image Selection. The image selection view, shown in Figure 5 enables analysts to navigate images by frame and experimental conditions. Each row displays the location id and an array of ticks, where each tick mark corresponds to an image recorded at that location. Clicking on the tick loads that image. The size of the bar on the left and the size of the tick marks are scaled based on the number of cells at the location or the image.

Small circles above the tick marks indicate that an exemplar has been extracted from that frame. A pair of larger circles above and below a frame indicates the currently selected frame. As cells are extracted into tracks, we show an arc from the first to the last time a cell appears in a track for selected cells.

Image Viewer. The image viewer (Figure 6) shows the raw image data of the cells along with color-coded segmentation boundaries. Red boundaries indicate that a cell is part of the current selection, blue are not. Green cells have been filtered out of the current data set. Metadata about a cell is shown in a tool-tip. The images can be inverted, as illustrated in Figure 6 —— which can result in better discriminability between cells and background for certain datasets. Images over frames can be navigated with keyboard cursors.
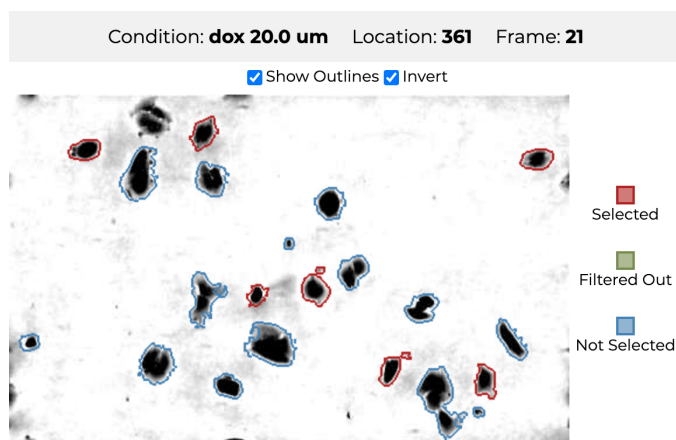
Fig. 6: The image viewer, showing a color-inverted image and segmentation outlines. The segmentation outlines are color-coded by whether they are selected or filtered out.
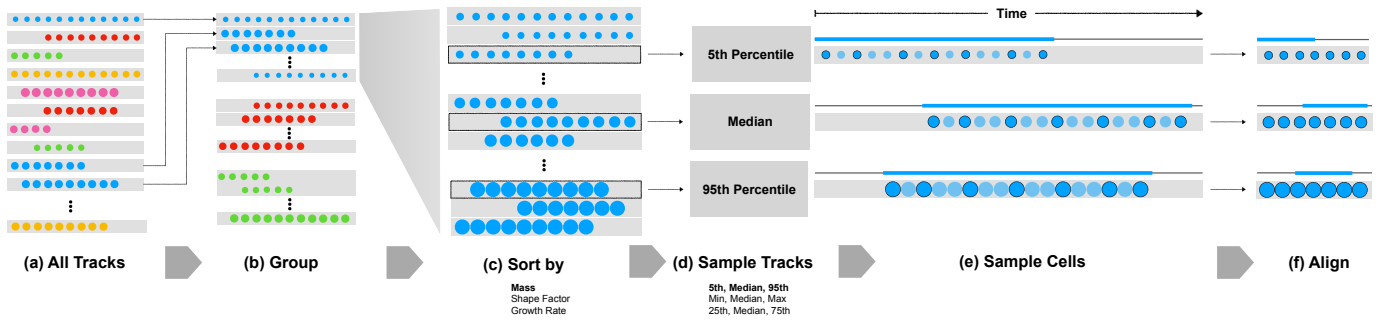
Fig. 7: The process of sampling exemplar cells based on a user-selected attribute. (a) Tracks with cells as extracted during preprocessing. (b) We first group cells by conditions, indicated as colors. (c) Next, we sort within each group by a user-selected attribute. We use mass in this example, but others such as growth rate or a shape metric are equally useful. (d) We then sample tracks based on a user-specified rule. In this example, we extract the median track, and the tracks at the 5th and 95th percentile according to the metric specified in the previous step. (e) Optionally, tracks can also be sampled over time. (f) The sampling then makes it possible to align the cells. The "time-window" of the track is shown above the cells with a filled-in bar.

Visualizing Exemplars of Cells. While the images that can be accessed via the selection view and analyzed in the image view, doing so is not scalable, neither for quality control nor for data analysis task. To address this problem, we introduce exemplar cells which we sample from the large set of cells based on user-chosen attributes. Exemplar cells are extracted from the original image based on their cell region to create little thumbnails. While exemplars of cells have been used before, [12, 18], we are the first, to our knowledge, who sample exemplars systematically based on a user-chosen attribute.

Figure 7 illustrates the process. Given cells that have been assigned to tracks in a preprocessing step, we first group the tracks by condition. Then we sort within each group by a user-specified attribute. The attributes can be cell-based, such as the mass of a cell, or track based, such as the growth of a cell over time. Next, we sample the tracks, for example, to show the median track and the tracks at the 5th and 95th percentile, thereby reducing all tracks for a group to a few representatives: a typical example and two outliers, in this case. Finally, in an optional step, we also sample exemplars over time, so that we always have the same number of cells for each track, no matter how long the track is. While this results in a desirable compact view, understanding the length of a track is an important aspect of the analysis, hence we

encode the time-frame over which the track is present in a bar on top of the exemplars.

Figure 8 shows an example of how this concept is implemented in Loon and introduces several complementary views. By default, seven images are evenly sampled from the cell's lifetime, though analysts have the option to show all of the cells in a track. The extent of a track's lifetime is visualized with the color-coded line above each track. On the left of the exemplars associated with a condition, we show a histogram that visualizes the distribution of the tracks with respect to the user-selected attribute (average mass in this case). A set of pins indicate the position of the exemplars in that histogram. Analysts can also retrieve exemplars at any location of that histogram; an example for a very large cell group is shown in Figure 8, indicated by the push-pin. This feature enables analysts to interactively check tracks anywhere in the distribution, e.g., to analyze outlier cells (**DA-4**), or to check the segmentation in edge cases (**QC-1**). Finally, a line chart to the right of each group shows the average growth rate for the condition and the growth rates of the exemplar cells. This view enables analysts to judge which cells might be responsible for growth or shrinkage. Analysts can also bookmark cells they encounter at any point in their analysis, which are then shown as exemplars at the top.
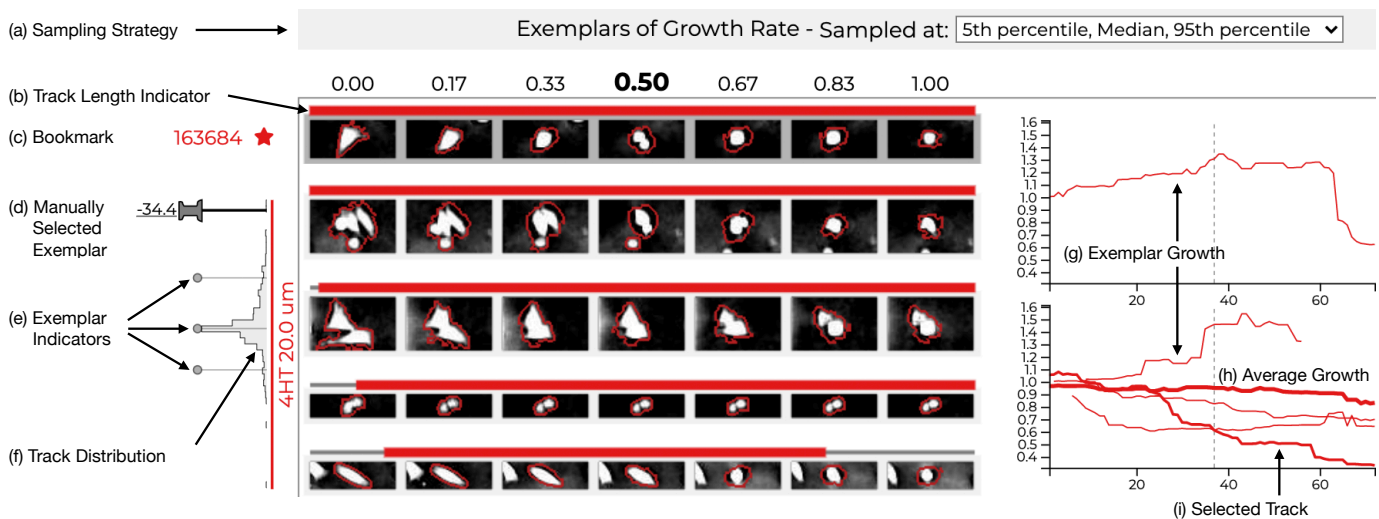
Fig. 8: Visualizing Exemplars. The exemplar view shows exemplar tracks, sampled from all tracks based on a user defined attribute and sampling strategy. (a) This example is sampled by the average mass and shows three tracks by default. (f) To the left of the exemplars, we show a histogram visualizing the distribution of tracks for the selected attribute (mass in this case). (e) Pins indicate the position of the exemplar tracks in this distribution. (d) Analysts can also click the histogram to manually select an exemplar and add it to the view. Such exemplars are shown as a push-pin in the histogram. (b) Colored bars on top of the exemplar tracks indicate the length of the track. A line chart on the right shows the (h) average growth of the condition, and (g) the growth rates of the exemplars. (b) Finally, any cell can be bookmarked. Bookmarked cells are shown as exemplar on top, independent of their group/condition. Several cells here are shrinking in mass. The pinned track (d), drawn from the extreme end of the distribution, shows an example of a cell cluster breaking apart, which contributes to this extreme drop in mass (i). All other cancer cells seem to be shrinking naturally.

Fig. 9: An early attempt to visualize tracks of cells without sampling.

We found that this condensed display of exemplar cell tracks is on a sweet-spot with regards to spatial efficiency and supporting the goals of our collaborators. In earlier iterations, we explored other ways to present these cell tracks, which are still accessible in the interface and may be useful in special cases. One alternate mode samples tracks but does not sample cells over time. We quickly found that the temporal sampling rate is too fine-grained and that cells mostly don't change much between just two frames, hence a view like this shows too much detail for most analysis tasks. We also initially attempted to visualize all cells in all tracks, aligned by absolute time, as shown in Figure 9. This alignment leads to more whitespace and tedious scrolling, as well as a significant slow-down with regards to the rendering performance.

6.4 Interaction and Metadata Views

Various domain tasks rely on the analysis of metadata, including quality control (**QC-3**), data cleaning (**DA-1**), and analysis (**DA-3**. Our example datasets, contains metrics for mass, size, as well as a shape factor that describes the "circularity" of a cell, in addition to track-level attribute such as growth.

Loon provides several standard statistical plots, such as scatter-plots, histograms, kernel-density estimate plots, and dot plots with box plots overlaid, along with a data selection interface, to visualize these data dimensions (see Figure 10a). Loon also provides the ability to facet each of these plots by condition.

In addition to showing the data, these plots serve as a selection and filtering interface. Using 1D and 2D brushes will update all other plots to indicate the current selection. For example, Figure 10a shows a brush made on the growth rate, which is also applied to the growth curves in Figure 10b as overlaid curves. At any point, a selection can be converted into a more permanent filter. Filters completely remove data outside the current selection. Importantly, this conversion will only refer to selections at the track level. In other words, it will not filter out parts of a track; tracks are either fully included or excluded.

7 IMPLEMENTATION

Loon is implemented as an open-source web-based tool. The source code is available at `https://github.com/visdesignlab/loon/`. Loon uses a Flask server as its backend. The various charts were custom-built with the D3 JavaScript library [5].

Our collaborators store their processed data, including the images and derived datasets, as MATLAB files on Google Drive, which provides free unlimited storage to educational organizations. Since a single experiment can produce in excess of 500GB of data, and the goal is to eventually run experiments daily, we were hesitant to duplicate this data on a dedicated server. Instead, we decided to dynamically query the data through the Google Drive API, which, however, is problematic regarding the performance of retrieving images. To address this, we implemented several strategies. First, we found that for visual representation we do not need the lossless image files. Slight downscaling and conversion to JPG reduced the size by about 100-fold. Next, we introduce three levels of caching. First, we cache all metadata on our webserver and automatically update it if the data on Google Drive has changed. Second, we use the IndexedDB Web API to store data on a web client for both image metadata and the images. To minimize the amount of data transferred, this caching is done dynamically as data/images are requested. A downside of this approach is that an initial load of the page will take on the order of minutes, but a reload will be on the order of seconds. Similarly, actions inside the tool that require more images will be slower, but reloads with the same data are fast. The final level of caching occurs in JavaScript memory for images.

(a) Metadata Views

(b) Effect of Brushes.
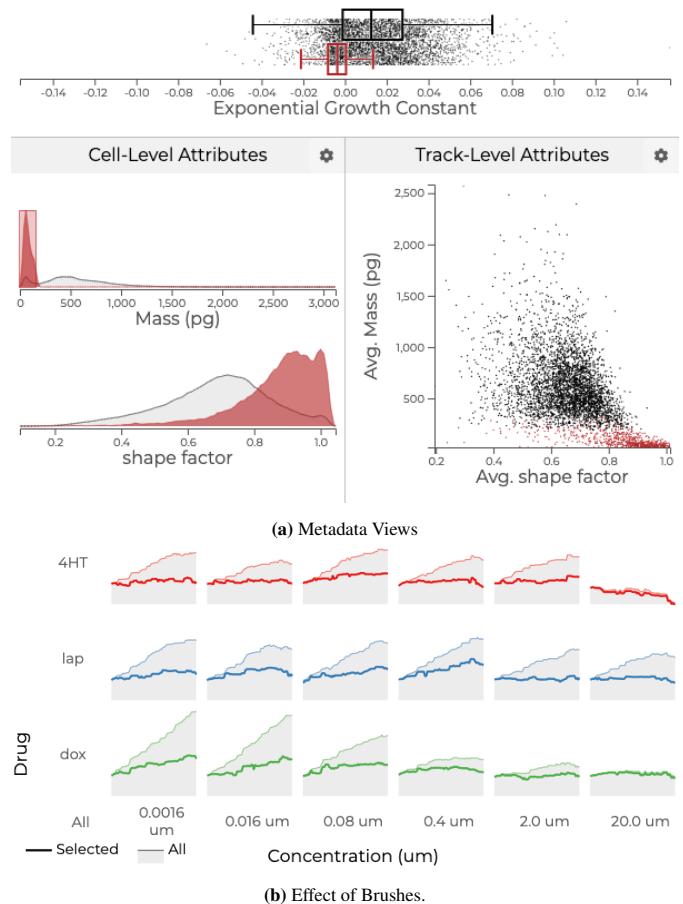
Fig. 10: Metadata views and selections. (a) Histograms, dot plots, and KDE plots visualize metadata. A brush on growth rate is reflected in all other plots, including in (b) condition selection views' growth curves. Here, a new growth curve, only for the selected elements is overlaid.

Since some image processing/extraction happens on the fly, we use image Blob objects and keep a set number in memory. This allows us to achieve frame navigation at rates that appear to animate cell growth, for instance.

8 CASE STUDIES

To demonstrate the utility of our tool, we provide two case studies, which were conducted by domain scientists and with real data and real analysis problems [32]. Our collaborators, however, were closely involved in the design of Loon and are also co-authors of this paper. Given issues with experimental demand characteristic effects that arise in close collaborations [6], we refrain from reporting subjective assessments of our collaborators and rather report factually on two analysis scenarios and how they were conducted with Loon, and describe the difference in capabilities relative to our collaborator's previous process.

It is also worth noting, that our collaborators used successive iterations of Loon over 18 months and co-developed their analysis pipeline to both provide all the necessary data items for Loon and to address analysis problems that were discovered when using Loon. For example, when visualizing the physical location of cells, our collaborators found positions far outside the expected bounds as well as points with a shape factor above one, when one should be the maximum (**QC-3**). This turns out to be an error and needs to be corrected in the pipeline.

8.1 Comparing Drug Effects on a Breast Cancer Case

We report here on an analysis of the breast cancer dataset that was conducted by Dr. Zangle, while the first authors observed remotely and recorded the session. A detailed record of all stages of the case study is shown in the supplementary material. Initially, he viewed the

normalized mass growth curves in the condition selection matrix (**DA-3**). There he identified a dramatic change for the drug 4HT between 2 and 20μM (see Fig 3). He commented that this appears to indicate that 4HT is cytotoxic at 20μM, but has little to no effect at lower concentrations. However, it is possible that the cells with negative growth at concentration 20μM are not actually dying, but that this is a segmentation or tracking problem instead (**QC-1**, **QC-2**).

To explore this question. Dr. Zangle selected 4HT at concentration 20μM (**DA-2**). Looking first at the single image view, he could identify cells that appeared to be dying. After bookmarking a particularly interesting cell, he confirmed with the mass over time curve that the cell was dying at the end of the track (**DA-3**), as shown in Figure 8.

Next, he changed the track attribute to view exemplars sampled from the growth rate. Then, within the exemplar tracks view, he inspected various cells with negative growth rates from the automatically selected exemplars as well as by manually pinning other cells with negative growth rates from the histogram. While inspecting the negative mass over time plot, Dr. Zangle became curious to find out whether the cell is dying. To answer this question requires analyzing the image data and segmentation. In one case, Dr. Zangle viewed the segmentation of a cell cluster, shown in Figure 8, and noticed that one cell left drifted away from a cluster until it was no longer considered part of the same region by the algorithm. He noted, that this explains part of the dramatic cell mass decrease (**QC-1**). Other cell clusters, however, showed clear segmentation, negative growth rate, and the cell morphology clearly indicated cell death. All of this combined does seem to indicate that 4HT is cytotoxic at 20μM. When communicating this, for instance in a paper, our collaborators would include selected exemplar images to illustrate what is happening to cells exposed to that drug (**DA-5**).

Another question Dr. Zangle's team frequently asks is what is causing differences in the average growth rate between conditions. It could be due to every cell growing more slowly, or from a heterogeneous response, with some cells dying, while others are unaffected (**DA-4**). Dr. Zangle was interested in this question for 4HT at 2μM. Here we see a much more moderate decrease in average growth rate (**DA-3**). To begin analyzing the heterogeneity he selected three conditions, 4HT at 2μM, 0.0016μM, and DMSO, the control for this drug. Being interested in the effect of growth rate on the cells, he chose growth rate as the attribute driving the sampling of exemplars. In the exemplar tracks view, he first looked at the growth rate histograms for each of these conditions (**DA-4**). Between these three there was no significant difference in the shape of this distribution. Next, he changed the sampling strategy to include more exemplars per condition. These exemplars, along with some manually pinned exemplars, provided a small group of cells to inspect when looking for cell death. When trying to identify dying cells, Dr. Zangle looked at the image characteristics as well as the growth rate to determine if a cell is actually dying or not. After inspection of exemplars from these three conditions, there was no noticeable difference in the number of cells dying (**DA-4**). After this analysis, he concluded that the likely explanation is that these three conditions don't exhibit significant heterogeneity.

8.2 Refining The Tracking for a Melanoma Dataset

In recent years, great strides have been made in the treatment of metastatic melanoma — the deadliest of skin cancer. These include targeted therapies against common oncogenic mutations and immunotherapies. Both therapies can be effective, but neither works with 100% efficacy [3]. This is due to the heterogeneous nature of these types of cancer cell populations. Determining which therapy will ultimately be most successful in preventing metastasis requires a better understanding of the effect drugs have on these different subpopulations of cells. A functional precision medicine approach that predicts response to targeted therapies could aid oncologists in this difficult and critical decision. To answer these questions, Dr. Judson-Torres' lab is running QPI experiments on their commercial platforms.

They started a collaboration with Dr. Zangle's lab to also integrate their dataset with their analysis pipeline and with Loon, as they hope to better analyze Since this is a new dataset for our collaborators' data processing workflow a key initial task is quality control. This case
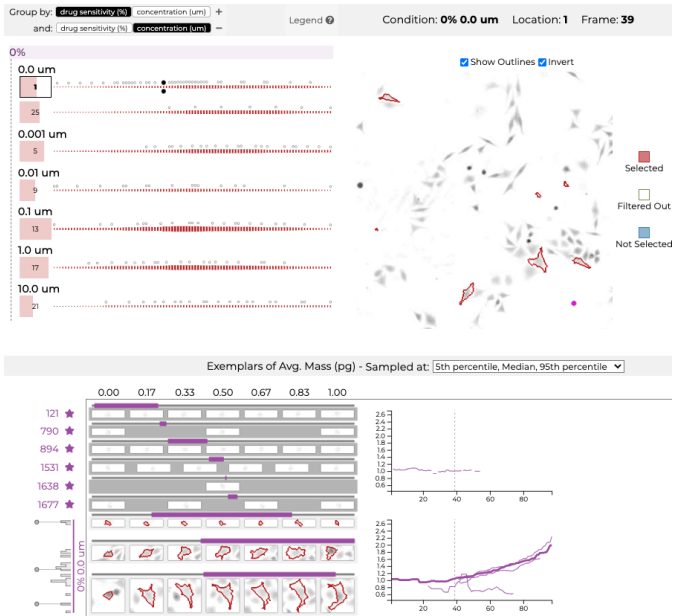
Fig. 11: Loon showing tracking problems in the melanoma dataset. The bookmarked short tracks should be merged into a single track.

study describes a multi-week process, where the lab analyzed the data, found deficiencies, refined it, etc.

As a first data cleanup step in Loon, Dr. Zangle filtered to only show longer tracks (**DA-1**). In the single image view, he noticed that only a few cells were being tracked for a long duration, as indicated by the color-coded outlines. Such short tracks are indicative of tracking problems. He then selected a few locations with the image selection view and referenced the tick sizes compared tick sizes across all locations to ultimately conclude that cells were commonly tracked only briefly (**QC-2**). To improve tracking, our collaborators have many parameters they can adjust. For instance, if a cell moves "too far", or changes mass "too much" in a single frame, it will not be tracked. This threshold of "too much" can be adjusted for different experiments. The image view is useful here since it shows the position and mass of these cell segments. Our collaborators use this to find cells that aren't tracked and then adjust their thresholds (**DA-1**). They use a similar threshold to distinguish between debris that has been segmented in the image and actual small cells.

Adjusting these thresholds improved the tracking for this dataset, but the improvement did run into an upper limit. When filtered to longer tracks, Dr. Zangle noticed that the cells in these tracks did not move much, whereas the cells that were not tracked well were more motile (**QC-2**). This particular dataset is challenging due to the variety of cell density and motility. He concluded that they would have to make larger changes to their tracking algorithm, possibly by connecting tracks after the first pass of the original algorithm.

For this QC task, our collaborators commented that the bookmarking feature was especially useful for keeping track of shorter tracks and comparing values of endpoints (**QC-2**). Figure 11 shows six tracks that ideally should be connected into a single one.

8.3 Loon Feedback

We asked both labs to briefly explain how Loon changes their workflow. Dr. Zangle responded:

> Loon enables us to view cell segmentation and morphology alongside mass over time tracks, which would be extremely tedious with our previous approach. This capability improves QC because we can quickly spot-check segmentation and find errors and how they interact with cell mass measurements. This also helps with interpreting data because we can see how cell morphology and motility are related to tracked parameters.

Dr. Judson Torres similarly commented:

> The workflow we find most useful in Loon is the ability to quickly visualize the mass, tracking information, and exemplar images of object groups based upon feature filters. The dry mass, track length, and, of course, the actual image of the segmented objects are all critical for conducting quality control — identifying which objects represent cells or colonies of interest and which are artifact from the segmentation. This process has previously been extremely tedious with our existing approaches, so much so that we had started to just accept a certain amount of noise in our data due to segmentation artifacts. In addition, one of our main uses of QPI approaches is to identify groups of phenotypically distinct cells within heterogenous populations. Our previous approaches first required analyses of feature-level data to identify distinct groups of objects followed by cross-referencing representatives of each group in the images to determine whether the observed feature clusters were real or artifact — a time intensive process. The ability to select groups based upon features and, in real time, visualize exemplars from each group enables us to rapidly identify and verify distinct groups of cells.

9 LESSONS LEARNED AND REFLECTIONS

Transferable Contribution. Even though we present a design study — a visualization tool that addresses the important but narrow problem of aiding in selecting cancer drugs based on microscopy experiments — we argue that our main technical contribution, sampling exemplars along selected data dimensions, transfers to a broad range of problems.

In general, we argue that visualizing through exemplars could potentially be applicable once two criteria are met. First, the dataset has to be large enough that it is infeasible to manually inspect every data point. Second, the data items have to contain rich contextual information that cannot be summarized. In our case, we consider our cell images as these data points. However, the same basic idea could be applied to many situations.

Obvious related applications that fulfill these criteria are in other image segmentation and microscopy scenarios, such as in connectomics or radiology, where both our patch-extraction and sampling could be applied. Different examples include time-series data, where periodic signals, such as heartbeats or electric data measured with an oscilloscope [25] are likely amenable to the same approach. Given the right metrics, even the analysis of cliques or other structures in a large network could benefit from exemplars [4]. Finally, an avid biker may be interested in looking back at their rides over the past few years. A traditional visualization approach would show distributions of ride duration, distance, and speed. An exemplar-based approach would pull out specific rides based on some criteria, such as distance traveled, then present entry points to inspect the entire ride. With the richer context, this provides our biker can observe different aspects of the data — like see a familiar bike route, or recall an exceptionally grueling ride.

Line Chart Design. Line charts are commonly considered a scalable solution for plotting multiple time series at the same time. However, we found that an overlaid line-chart was unsuitable when the goal was to make a series of comparisons based on categorical information associated with the lines, such as the drugs and concentrations in our case. Here, faceted small multiples were much more useful, especially as the faceting can be done based on the categorical variables.

Co-Design of Method and Visualization. This design study was unique in that it contained strong aspects of co-design between the visualization researchers and the domain experts. QPI for drug screening is a novel method, and the Zangle lab is at the forefront of developing this technology. As illustrated in the melanoma case study, the lab is not treating the visualization as an end-point, but rather as a means to understand their data **and** their process. Just as the visualization tool developed over this 18-month collaboration, so did

the analysis process. We found that the immersive approach we took over this time was especially valuable in this context.

10 DISCUSSION

Performance. Loon can be slow when a dataset is used for the first time. Even though we implement various caching strategies (see Section 7), there is still a lot of data to be transferred from a relatively slow data source. To address this problem, we will likely have to abandon our initial design goal of retaining the data on our collaborator's Google Drive so that Loon integrates nicely with their workflow. Image-pyramid based approaches have been successfully used for microscopy visualization [26] and combined with other pre-computations and efficient data structures, we expect to achieve significantly improved performance.

Sampling Strategies for Exemplars. Our current sampling strategies for selecting exemplars are simple. This has the advantage that they are easily understood and explainable. However, we believe that it would be valuable to add more sophisticated sampling strategies, based on multiple attributes at once, or maybe using the depth measure from contour boxplots [34] to sample based on growth functions. We could also use representative examples based on clustering algorithms, or extreme examples based on archetypal analysis [10].

Evaluation. We considered other evaluation strategies, in addition to our case studies, that demonstrate the value of our contribution. However, we came to the conclusion that other approaches would be problematic with respect to their validity, or test aspects that are not relevant to our contribution. While, for example, usability testing would be useful to identify usability problems and bugs, it does not result in generalizable knowledge about our contribution. A comparative evaluation, either based on insights or quantitative measures is infeasible with a specialized tool, as the number of users is very limited and participants will possibly be biases since they know the experimenter and the desired outcome. A user satisfaction survey likely suffers from similar biases. Hence, we chose case studies to demonstrate that our tool is useful for the domain goals we claim and use careful arguments to justify our technical contribution. However, in the future, we believe that an empirical evaluation of our main technical contribution (exemplars) could be designed, yet this would require carefully untangling our approach to exemplars from the larger interface. We argue that case studies are a good first approach to validate that the idea is compelling, but that an empirical study of exemplars and alternatives is beyond the scope of this work and should be conducted systematically by an outside party [29].

11 CONCLUSION

In this paper, we presented Loon, a visualization tool for analyzing the effectiveness of cancer drugs on individual cell lines based on QPI microscopy data. In addition to the tool, we also contribute the concept of analyzing large datasets with complex item characteristics by exemplars. We argue that our tool is a successful design study in a high-impact area and that our technical contribution transfers to various other scenarios.

We plan on continuing our collaboration with the domain scientist to expand and refine Loon. One example of potential future directions is the integration of lineage diagrams, which track cells that divide over time [15]. Such diagrams would aid in identifying cell death and heterogeneous growth.

REFERENCES

[1] A. K. Ai-Awami, J. Beyer, D. Haehn, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger. NeuroBlocks – Visual Tracking of Segmentation and Proofreading for Large Connectomics Projects. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):738–746, Jan. 2016. doi: 10.1109/TVCG.2015.2467441

[2] G. Anselmi and M. Humphry. Livecyte: Creating a comprehensive cell profile. *Microscopie*, 30(2), 2019. doi: 10.4081/microscopie.2019.8592

[3] P. A. Ascierto, L. Borgognoni, G. Botti, M. Guida, P. Marchetti, S. Mocellin, P. Muto, G. Palmieri, R. Patuzzo, P. Quaglino, I. Stanganelli, and C. Caracò. New paradigm for stage III melanoma: From surgery to adjuvant treatment. *Journal of Translational Medicine*, 17(1):266, Aug. 2019. doi: 10.1186/s12967-019-2012-2

[4] B. Bach, N. Henry-Riche, T. Dwyer, T. Madhyastha, J.-D. Fekete, and T. Grabowski. Small MultiPiles: Piling Time to Explore Temporal Patterns in Dynamic Networks. *Computer Graphics Forum*, 34(3):31–40, June 2015. doi: 10.1111/cgf.12615

[5] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '11)*, 17(12):2301–2309, 2011. doi: 10.1109/TVCG.2011.185

[6] B. Brown, S. Reeves, and S. Sherwood. Into the wild: Challenges and opportunities for field trial methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pp. 1657–1666. Association for Computing Machinery, New York, NY, USA, May 2011. doi: 10.1145/1978942.1979185

[7] A. E. Carpenter, T. R. Jones, M. R. Lamprecht, C. Clarke, I. H. Kang, O. Friman, D. A. Guertin, J. H. Chang, R. A. Lindquist, J. Moffat, P. Golland, and D. M. Sabatini. CellProfiler: Image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10):R100, Oct. 2006. doi: 10.1186/gb-2006-7-10-r100

[8] S. Carter, Z. Armstrong, L. Schubert, I. Johnson, and C. Olah. Activation Atlas. *Distill*, 4(3):e15, Mar. 2019. doi: 10.23915/distill.00015

[9] J. Chun, T. A. Zangle, T. Kolarova, R. S. Finn, M. A. Teitell, and J. Reed. Rapidly quantifying drug sensitivity of dispersed and clumped breast cancer cells by mass profiling. *Analyst*, 137(23):5495–5498, 2012. doi: 10.1039/C2AN36058F

[10] A. Cutler and L. Breiman. Archetypal Analysis. *Technometrics*, 36(4):338–347, Nov. 1994. doi: 10.1080/00401706.1994.10485840

[11] D. Dao, A. N. Fraser, J. Hung, V. Ljosa, S. Singh, and A. E. Carpenter. CellProfiler Analyst: Interactive data exploration, analysis and classification of large biological image sets. *Bioinformatics*, 32(20):3210–3212, Oct. 2016. doi: 10.1093/bioinformatics/btw390

[12] K. Dinkla, H. Strobelt, B. Genest, S. Reiling, M. Borowsky, and H. Pfister. Screenit: Visual Analysis of Cellular Screens. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):591–600, Jan. 2017. doi: 10/f92c98

[13] A. A. Friedman, A. Letai, D. E. Fisher, and K. T. Flaherty. Precision medicine for cancer with next-generation functional diagnostics. *Nature Reviews Cancer*, 15(12):747–756, Dec. 2015. doi: 10.1038/nrc4015

[14] N. A. Hamilton, J. T. Wang, M. C. Kerr, and R. D. Teasdale. Statistical and visual differentiation of subcellular imaging. *BMC Bioinformatics*, 10(1):94, Mar. 2009. doi: 10.1186/1471-2105-10-94

[15] G. Hattab, V. Wiesmann, A. Becker, T. Munzner, and T. W. Nattkemper. A Novel Methodology for Characterizing Cell Subpopulations in Automated Time-lapse Microscopy. *Frontiers in Bioengineering and Biotechnology*, 6, 2018. doi: 10.3389/fbioe.2018.00017

[16] M. Hejna, A. Jorapur, Y. Zhang, J. S. Song, and R. L. Judson. Quantification of mammalian tumor cell state plasticity with digital holographic cytometry. In *Quantitative Phase Imaging IV*, vol. 10503, p. 1050312. International Society for Optics and Photonics, Feb. 2018. doi: 10.1117/12.2290462

[17] K. Holtzblatt and H. Beyer. Contextual Design: Evolved. *Synthesis Lectures on Human-Centered Informatics*, 7(4):1–91, Oct. 2014. doi: 10.2200/S00597ED1V01Y201409HCI024

[18] T. R. Jones, I. H. Kang, D. B. Wheeler, R. A. Lindquist, A. Papallo, D. M. Sabatini, P. Golland, and A. E. Carpenter. CellProfiler Analyst: Data exploration and analysis software for complex image-based screens. *BMC Bioinformatics*, 9(1):482, Nov. 2008. doi: 10.1186/1471-2105-9-482

[19] E. Jurrus, S. Watanabe, R. J. Giuly, A. R. C. Paiva, M. H. Ellisman, E. M. Jorgensen, and T. Tasdizen. Semi-Automated Neuron Boundary Detection and Nonbranching Process Segmentation in Electron Microscopy Images. *Neuroinformatics*, 11(1):5–29, Jan. 2013. doi: 10/f4k492

[20] E. Kerzner, S. Goodwin, J. Dykes, S. Jones, and M. Meyer. A Framework for Creative Visualization-Opportunities Workshops. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):748–758, Jan. 2019. doi: 10.1109/TVCG.2018.2865241

[21] E. Kerzner, A. Lex, C. L. Sigulinsky, T. Urness, B. W. Jones, R. E. Marc, and M. Meyer. Graffinity: Visualizing Connectivity in Large Graphs. *Computer Graphics Forum (EuroVis)*, 36(3):251–260, 2017. doi: 10.1111/cgf.13184

[22] I. A. Khan, M. Lupi, L. Campbell, S. C. Chappell, M. R. Brown, M. Wiltshire, P. J. Smith, P. Ubezio, and R. J. Errington. Interoperability of time series cytometric data: A cross platform approach for modeling tumor heterogeneity. *Cytometry Part A*, 79A(3):214–226, 2011. doi: 10.1002/cyto.a.21023

[23] R. Krueger, J. Beyer, W.-D. Jang, N. W. Kim, A. Sokolov, P. K. Sorger, and H. Pfister. Facetto: Combining Unsupervised and Supervised Learning for Hierarchical Phenotype Analysis in Multi-Channel Image Data. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):227–237, Jan. 2020. doi: 10.1109/TVCG.2019.2934547

[24] F. Lekschas, B. Bach, P. Kerpedjiev, N. Gehlenborg, and H. Pfister. HiPiler: Visual Exploration of Large Genome Interaction Matrices with Interactive Small Multiples. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):522–531, Jan. 2018. doi: 10.1109/TVCG.2017.2745978

[25] R. Lopez-Hernandez, D. Guilmaine, M. J. McGuffin, and L. Barford. A layer-oriented interface for visualizing time-series data from oscilloscopes. In *2010 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 41–48, Mar. 2010. doi: 10.1109/PACIFICVIS.2010.5429607

[26] T. Manz, I. Gold, N. H. Patterson, C. McCallum, M. S. Keller, B. W. H. Ii, K. Börner, J. M. Spraggins, and N. Gehlenborg. Viv: Multiscale Visualization of High-Resolution Multiplexed Bioimaging Data on the Web. 2020. doi: 10.31219/osf.io/wd2gu

[27] M. Meyer, T. Munzner, A. DePace, and H. Pfister. MulteeSum: A Tool for Comparative Spatial and Temporal Gene Expression Data. *IEEE Transactions on Visualization and Computer Graphics (InfoVis '10)*, 16(6):908–917, 2010. doi: 10.1109/TVCG.2010.137

[28] M. Meyer, B. Wong, M. Styczynski, T. Munzner, and H. Pfister. Pathline: A Tool For Comparative Functional Genomics. *Computer Graphics Forum (EuroVis '10)*, 29(3):1043–1052, 2010. doi: 10.1111/j.1467-8659.2009.01710.x

[29] C. Nobre, D. Wootton, L. Harrison, and A. Lex. Evaluating Multivariate Network Visualization Techniques Using a Validated Design and Crowdsourcing Approach. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, 2020. doi: 10.1145/3313831.3376381

[30] A. J. Pretorius, I. A. Khan, and R. J. Errington. A Survey of Visualization for Live Cell Imaging. *Computer Graphics Forum*, 36(1):46–63, 2017. doi: 10.1111/cgf.12784

[31] M. Sebesta, P. J. Egelberg, A. Langberg, J.-H. Lindskov, K. Alm, and B. Janicke. HoloMonitor M4: Holographic imaging cytometer for real-time kinetic label-free live-cell analysis of adherent cells. In *Quantitative Phase Imaging II*, vol. 9718, p. 971813. International Society for Optics and Photonics, Mar. 2016. doi: 10.1117/12.2216731

[32] M. Sedlmair, M. Meyer, and T. Munzner. Design Study Methodology: Reflections from the Trenches and the Stacks. *IEEE Transactions on Visualization and Computer Graphics (InfoVis)*, 18(12):2431–2440, 2012. doi: 10.1109/TVCG.2012.213

[33] T. Walter, D. W. Shattuck, R. Baldock, M. E. Bastin, A. E. Carpenter, S. Duce, J. Ellenberg, A. Fraser, N. Hamilton, S. Pieper, M. A. Ragan, J. E. Schneider, P. Tomancak, and J.-K. Hériché. Visualization of image data from cells to organisms. *Nature Methods*, 7(3):S26–S41, Mar. 2010. doi: 10.1038/nmeth.1431

[34] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, Dec. 2013. doi: 10.1109/TVCG.2013.143

[35] T. A. Zangle and M. A. Teitell. Live-cell mass profiling: An emerging approach in quantitative biophysics. *Nature Methods*, 11(12):1221–1228, Dec. 2014. doi: 10.1038/nmeth.3175

[36] H. Zeng, A. Jorapur, A. H. Shain, U. E. Lang, R. Torres, Y. Zhang, A. S. McNeal, T. Botton, J. Lin, M. Donne, I. N. Bastian, R. Yu, J. P. North, L. Pincus, B. S. Ruben, N. M. Joseph, I. Yeh, B. C. Bastian, and R. L. Judson. Bi-allelic Loss of CDKN2A Initiates Melanoma Invasion via BRN2 Activation. *Cancer Cell*, 34(1):56–68.e9, July 2018. doi: 10.1016/j.ccell.2018.05.014