Published in final edited form as:

Nat Protoc. 2015 March ; 10(3): 459–474. doi:10.1038/nprot.2015.022.

Three-Dimensional Computational Reconstruction of Tissues with Hollow Spherical Morphologies using Single-Cell Gene Expression Data

Robert Durruthy-Durruthy^{#1}, Assaf Gottlieb^{#2}, and Stefan Heller¹

¹Department of Otolaryngology - Head & Neck Surgery, Stanford University School of Medicine, Stanford, CA 94305, USA

²Department of Genetics, Stanford University School of Medicine, Stanford, CA 94305, USA

[#] These authors contributed equally to this work.

Abstract

Single-cell gene expression analysis has contributed to a better understanding of the transcriptional heterogeneity in a variety of model systems, including those used in research in developmental, cancer, and stem cell biology. Nowadays, technological advances facilitate the generation of large gene expression datasets in high-throughput format. Strategies are needed to pertinently visualize this information in a tissue–structure related context, so as to improve data analysis and aid the drawing of meaningful conclusions. Here we describe an approach that utilizes spatial properties of the tissue source to enable the reconstruction of hollow sphere–shaped tissues and organs from single-cell gene expression data in three-dimensional space. To demonstrate our method, we used cells of the mouse otocyst and the renal vesicle as examples. This protocol presents a straightforward computational expression analysis workflow and is implemented on the MATLAB and R statistical computing and graphics software platforms. Hands-on time for typical experiments can be less than 1 h using a standard desktop PC or Mac.

Keywords

spatial transcriptomics; in situ hybridization; principal component analysis; dimension-reduction technique; otocyst; renal vesicle

INTRODUCTION

Measurement of gene expression levels at single-cell resolution has proven to be an accurate tool for in-depth analyses of cellular differentiation¹ and cancer development², and in

Correspondence to Stefan Heller, phone: 650-799-0594, fax: 650-721-2163.

E-mail addresses:

R.D.-D.: durruth1@stanford.edu

A.G.: assafgo@stanford.edu

S.H.: hellers@stanford.edu

Contributions R.D.D. and S.H. designed the experiments. R.D.D. performed the experiments. R.D.D. and A.G. performed data analysis. R.D.D., A.G. and S.H. wrote the manuscript.

Competing financial interests S.H. is affiliated with Inception 3, Inc. The rest of the authors declare no competing financial interests.

describing regulatory mechanisms of cell fate decision processes³. Typically, multiplex quantitative RT-PCR (qRT-PCR) or whole-transcriptome analysis (RNA-Seq) of individual cells can be conducted for many genes simultaneously, resulting in the accumulation of data that is complex and information-rich. Existing analysis strategies generally utilize basic differential gene expression calculations⁴, different clustering algorithms⁵, and dimension-reduction procedures⁵. Here, we describe a technique that amalgamates single-cell expression data with geometric modeling in the three-dimensional (3D) space. We utilize two expression datasets, one derived from qRT-PCR studies of 267 individual cells from the mouse otocyst, the precursor of the vertebrate inner ear⁶ and the second from RNA-Seq analysis of 57 cells from the renal vesicle, the first polarized epithelial precursor of the nephron⁷. The otocyst is a transient structure in early inner ear development that has the morphology of a hollow sphere, similar to the renal vesicle. An envelope-like arrangement serves as a template for the overall organization of individual cells in both tissues. Genes with previously known spatial expression domains are used to calculate a blueprint that recapitulates the position of each cell in the context of an orthogonal coordinate system representing the major organ axes. This platform provides a novel way to efficiently communicate biological data, and it enables the analysis of multidimensional data in an accessible and informative format.

Applications of the method

This protocol employs either qRT-PCR or RNA-Seq data collected from single cells and provides instructions on how to analyze and visualize the data in a representative 3D model of the analyzed spherical organ. The study of gene expression in individual cells affords a more accurate representation of cell-to-cell variations, in contrast to bulk measurements that only reflect the stochastic average of such expression⁸. A number of multivariate analysis techniques have been proposed to scrutinize large sets of data. Most of the existing approaches deploy various clustering algorithms to describe potential subpopulations and apply dimension-reduction protocols, such as principal component analysis (PCA), to resolve patterns of shared transcriptional identities^{9,10}. Although these methods are fundamental tools in generally organizing the biological heterogeneity of tissue-derived cells, they fail to take into account the original spatial organization of the tissue or organ. Gene expression data that reflect spatially encrypted properties can provide much deeper insight in the effort to understand cellular processes than gene expression information alone. The control and function of many cellular processes are tightly linked to and affected by the cells' spatial distribution¹¹. Single-cell gene expression analysis methods generally require the complete dissociation of tissues and organs. As a consequence, once individual cells are dissociated, spatial information is lost. The method described in this protocol enables the recovery of spatial information. Computational reconstruction of multicellular structures in which individual cells have defined positional parameters and assignment of gene expression values on a cell-to-cell basis presents an invaluable tool to characterize cellular identities in the context of their (micro)-environment. Our protocol provides geometric modeling of shell-like, spherically shaped organs and tissues by integrating gene expression data derived from single cells with computational dimension-reduction methods, such as PCA.

This protocol can generally be applied to a variety of tissues that fit the morphological requirements, including the otocyst, renal vesicle, optic vesicle¹², seminal vesicle¹³, or Kupffer's vesicle¹⁴ to name only a few. We note that the native cellular organization can vary between species and developmental stages. Particularly, spherical organs with more than one cellular layer, such as the blastocyst, probably fall short of being adequately represented with this protocol. Nevertheless, we envision that — given the appropriate numerical equations — this method can be expanded and used widely to comprehensively reconstruct sphere-shaped tissues and organs based on quantitative mRNA expression data of individual cells, in high-throughput and high-resolution.

Comparison with other methods

In-situ hybridization and immunohistochemistry enable expression analysis of mRNA and proteins. 3D reconstructions of expression domains identified with these methods can be conducted using stacks of microtome or optical sections¹⁵. However, the throughput of these 3D reconstructions is low and only a few genes can be tested in parallel. On the other hand, microarray platforms as well as population-based RNA deep sequencing techniques enable the simultaneous measurements of thousands of genes¹⁶, yet these measurements are generally performed in bulk cell populations. Our method combines the benefits of high-throughput gene expression data acquisition with restored spatial information at single-cell level (Figure 1). The resulting technology enables the analysis of intricate gene expression data within the 3D context of multicellular systems.

To our knowledge, no comparable techniques exist that utilize single-cell gene-expression data to generate a comprehensive, spatially delineated expression atlas in a quantitative manner. SINGuLAR, a computational platform based on the statistical programming language R, developed by Fluidigm, enables analysis of large-scale quantitative expression data using techniques such as PCA, hierarchical clustering, and violin plot diagrams; yet none of these techniques acknowledge the structural context of the cell-derived anatomical configuration, and they present the data in 2D format only (<http://www.fluidigm.com/singular-analysis-toolset.html>). Other approaches have been developed that rely on various image acquisition and computer-based reconstruction protocols and do not directly measure RNA levels in individual cells¹⁷. More recently, techniques have been developed to measure the RNA complement of the genome within cells of intact tissues with subcellular resolution¹⁸.

Possible future applications of the Protocol

Structures that are morphologically more sophisticated than those covered in the Procedure can similarly be computed and their geometric modeling is only limited by the availability of appropriate mathematical calculations and equations to describe the object in the 3D space; these approaches are, however, not covered in the present protocol. Because the anatomical characterization by mathematical equations constitutes an integral component of the protocol, we note that the geometric formulation should always be carefully examined and revised on an experiment-to-experiment basis. Furthermore, the protocol highlights the visualization aspect of single cell data and provides simple subsequent quantitation measures. We anticipate that the Procedure will serve as a basis for the research community

to further develop our approach, so that the practicality of it will be improved and refined. Such improvements and refinements include the implementation of alternative mathematical equations that enable the description of non-spherically-shaped and hollow organs in 3D space, adjustment of analytical parameters that are tissue-specific and data-collection-platform dependent, as well as the incorporation of functional features that aid in the quantitation process, such as the calculation of mean gene expression per anatomical domain. We envision that computational reconstruction of multicellular structures using single-cell gene expression data, coupled with tissue-contextual, automated quantitation features will transform the field of cell and developmental biology.

Limitations

The chances of success of this protocol are highly dependent on both the gene selection (in case of qRT-PCR) and feasibility of mathematically describing the cellular structure of a tissue or organ in 3D space (Figure 2). The fact that *a priori* knowledge about a minimum set of genes is necessary to implement this protocol represents a challenge in some instances, as satisfactory expression data is not always available (Figure 3). Therefore, the Procedure may not always be readily applicable and traditional initial validation experiments (e.g., *in situ* hybridization) are recommended to address such deficits. We also note that the present Protocol is not readily applicable to specimens derived from tumor samples. In these specimens, transcriptional heterogeneity is often accompanied by major morphological differences that vary from case to case. Additionally, the lack of a structured organization, which can be typically found in developing organs, exacerbates the difficulty of establishing axis-confined domains and makes this approach rather impracticable.

The methodology described herein exploits PCA as the underlying core technique. PCA is a mathematical approach to reduce the dimensionality of a multivariable dataset by concurrently retaining as much information as possible¹⁹ (Figure 3b). The result is a transformed coordinate system in which newly identified variables (the principal components) are arranged as linear combinations of the original variable vectors (the genes). However, expression profiles do not always conform to a linear relationship between genes in which case conclusive data visualization and interpretation may be difficult. Alternative techniques, such as non-linear PCA or other non-linear dimension reduction algorithms may help supplementing this guide, if more refined transformations are needed²⁰.

As with all applications that measure the abundance of RNA species, degradation effects can critically hamper the downstream analysis. We therefore remind the researcher to comply with appropriate RNA handling precautions. The present protocol assumes that proper and adequate quality control measures of raw data have been applied.

Experimental Design

Assay Design (Fig. 2); selection of anchor genes and control genes—Figure 2 summarizes the workflow of the present protocol, beginning with the determination of the tissue structure that one seeks to analyze, followed by the design of assays (required for qRT-PCR approaches only), and concluding with the *in silico* representation in 3D space and subsequent quantitation. Commonly single cell qRT-PCR approaches target the

transcriptome in a sequence-specific manner unlike RNA-Seq approaches where quantitative data of the *whole* transcriptome is measured. Therefore, experimental setups using qRT-PCR platforms require the design of specific primer pairs – called the assays – to amplify transcripts of interests. Naturally, this step is not necessary when using single cell RNA-Seq data. In experimental setups where qRT-PCR data is employed, the module 'Assay Design' consequently presents a vital stage in successfully applying the here-described methodology.

A careful and thorough selection of genes is crucial in effectively visualizing the data in 3D, as the expression data directly affects the degree of information that is preserved in the first principal components (Figure 3a,b). It is important to include genes with known expression domains that provide spatial information, permitting the delineation of putative organ or body axes, such as dorsal and/or ventral or the distal and/or proximal axes (Figures 4, 5). Therefore, prior knowledge of a finite number of genes, which we call 'anchor genes', with well-defined gene expression motifs is essential to exploit the full potential of this integrative technique. The inner ear dataset, utilized in the example application of the approach, relies on two anchor genes, with well-described expression domains in the otocyst, to determine all three axes of the organ structure in 3D space (Figure 6). The accuracy of this model was validated by confirmation or extension of the spatial expression of 35 additional genes with known expression in the mouse otocyst⁶. In the case of the nephron precursors, the analysis of which is also covered in the Procedure, we initially utilized twelve anchor genes to distinguish between proximal and distal domains. Of these twelve markers, eight qualified as conclusive anchor genes, similar to the conclusions reported by Brunskill *et al.*⁷

In addition to genes expected to be detectable in only a subset of cells, the list of assays should contain positive control genes with ubiquitous expression, such as *Actb* and *Gapdh*, as well as markers with expected absent expression (indicating off-target cells) for general quality control purposes. Depending on the collection method used for cell enrichment, the addition of representative negative markers is of particular importance. It will allow the researcher to exclude potential contaminating cells that are not part of the organ or tissue of interest and could interfere with the downstream analysis. Finally, any number of transcripts can be added for which expression patterns are unknown. With regard to the 267 otocyst-derived cells, the present protocol is based on the analysis of 96 genes of which about 40 had prior defined expression information for the organ⁶. Examples for genes with known expression patterns are *Oc90*, *Gbx2*, and *Lfng* that were predicted to be present in only a portion of all analyzed cells^{21–23}. *Actb*, *Gapdh* and enhanced green fluorescent protein (*Egfp*) reporter gene served as universally expressed control assays and *Pax6*, which is expressed in the hindbrain but not the otocyst, was used as a negative marker. For the present analysis of the 57 renal-vesicle-derived cells aimed at showing general applicability of the protocol, we included the following distal anchor genes as specified by the authors, Brunskill *et al.*⁷: *Pou3f3*, *Dll1*, *Sox9*, *Dkk1*, *Papss2*, *Greb1*, *Pcsk9*, *Lhx1*, *Bmp2*, as well as the proximal genes *Cdh6*, *Wt1*, *Tmem100*.

Single cell collection and data acquisition (Fig. 2)—A detailed description of experimental steps prior to ready-to-use data accumulation that include cell isolation and raw data acquisition is not provided in this protocol and can be found elsewhere^{24,25}. We

employed fluorescence-activated cell sorting (FACS) to collect individual cells from the target population (e.g., Pax2-Cre fate-labeled otocyst and neuroblasts of E10.5 mouse embryos)⁶. Similarly, nephron progenitor cells of P4 mice were assembled using FACS⁷. Alternative methods can be employed to enrich for the desired cells, such as magnetic cell sorting²⁶, laser capture microdissection²⁷ or micropipette aspiration²⁸. We also note that cell collection needs to be carried out in a timely manner. Pooling of cells that originate from different batches (i.e., embryos) has to be treated with caution to ensure that variation in gene expression indicates mainly spatial and not temporal differences^{6,29}. Regardless of the procedure applied, the total number of cells collected should correlate with measures of organ or tissue complexity, such as the expected number of different cell types of the tissue one seeks to geometrically model, as well as the tissue or organ's overall volume. The larger and more complex the multicellular structure of the tissue or organ, the more cells are necessary to adequately represent it computationally. It is reasonable to expect that an increase in the number of cells included in the analysis corresponds to an increased accuracy of the mathematical image and improved degree of gene expression map resolution. For RNA-Seq this increase in cell numbers used for analysis can be achieved cost-effectively by barcoding nucleic acids of individual cells to facilitate multiplexing of hundreds of cells for sequencing²⁹.

The present Protocol uses qRT-PCR data from 382 early inner ear cells generated on the Fluidigm Biomark HD platform⁶ from otocysts and neuroblast cells of mouse embryos. For 3D reconstruction, we focus on the 267 otocyst-derived cells identified by cluster analysis in Durruthy-Durruthy, R. *et al.*⁶. The RNA-Seq data from 57 early kidney cells was prepared using Fluidigm's C1 system³⁰ for automated reverse transcription and cDNA amplification as described in Brunskill E.W. *et al.*⁷. We note that 57 cells may not be an adequate number to representatively approximate the renal vesicle in its entirety.

Raw data-processing (Fig. 2)—This encompasses procedures like normalization and data outlier removal. To date, no widely agreed measure has been proposed to normalize single cell gene expression data. In approaches based on qRT-PCR, normalization techniques successfully applied are cell-based², gene-based¹, or both³. It should be the researchers' decision to evaluate different normalization techniques to determine which one is the most appropriate for their experimental paradigm. In the present protocol, in which qRT-PCR-derived data are utilized, we implemented cell-specific normalization factors that account for all genes across all cells, an approach we utilized in the study that is the basis of this Protocol⁶. In short, in Durruthy-Durruthy, R. *et al.*⁶, for every cell, the median Log2Ex value across all genes was calculated. The difference between this cell-characteristic value and the mean of all median Log2Ex values was subtracted from all Log2Ex values.

We note that it is the researcher's decision to define outlier cells, as this decision relies on various parameters (e.g., tissue type and collection method) and no universally valid approach exists. We assume that proper quality control measures are applied to assess the integrity of the raw data.

Bioinformatics Multivariate Analysis (Fig. 2)—We include this module to highlight the possible necessity to first identify putative distinct subpopulations that greatly differ

from each other with respect to their global transcriptional identity, suggesting dissimilar tissue site origins. In the case of the inner ear precursor cells, we filtered out a population of neuroblast cells that were not part of the otocyst cell population⁶.

Statistical analysis and visualization—In the Procedure, we provide instructions on how to implement the algorithm with two different computational environments, MATLAB (in the Procedure) and R (in Supplementary Methods). Alternative statistical analysis and visualization platforms can be used as well, such as Gnu Octave (<https://www.gnu.org/software/octave/>). We strongly recommend consulting documentations and manuals of MATLAB and R, as this protocol does not provide a comprehensive overview of how to use these software packages. More exhaustive and detailed resources for MATLAB can be found at <http://www.mathworks.com/help/matlab/> and http://www.mathworks.com/academia/student_center/tutorials/launchpad.html that include tutorials, videos, and Q&A sections. Tutorials for R are available at <http://cran.r-project.org/manuals.html>. Table 1 summarizes the MATLAB and R toolboxes used in this protocol. The present protocol contains updated algorithms than were used in the primary research publication⁶. We also want to remind the reader that suggested parameters depend on the tissue sample examined.

Level of expertise needed to implement the protocol—The step-by-step guide in the Procedure is designed for both the novice and expert in MATLAB or R. The goal is to provide a transparent and straightforward workflow that enables the researcher to address questions concerning single cell-associated gene expression data in the context of spatially construed features.

A general understanding of the morphology of the tissue from which the cells are isolated is required. Also, expression data for some genes (e.g., anchor genes) must exist that are necessary to (1) map putative body axes onto the model and (2) to validate the distribution of the cells using independent information. The herein mathematically described spherical characterization can be modified and adapted to different geometric systems such as cylindrical, cubical, and potentially more complex multi-segmental structures.

In order to familiarize themselves with the protocol, we suggest readers implement the Procedure using the example data provided and referred to below, using either of both available options (Step 2 and 3 option A versus Step 2 and 3 option B, see Procedure), before proceeding to analyze their own data.

Example Data—In the Procedure, to demonstrate the workflow, we employed the data set from our recent study⁶, in which we analyzed 382 single cells from the E10.5 mouse inner ear anlage consisting of 267 otocyst cells and 115 otic neuroblasts. In this protocol, we focus on the 267 cells of the otocyst. The numerical data is presented on a log-scale as Log2Ex values⁶. In addition we include the analysis of single-cell gene expression data that have been acquired using an RNA-Seq approach⁷. Here, cells are derived from P4 mouse renal vesicles and RPKM (reads per kilobase per million mapped reads) values were log2-transformed. Linear-scaled zero-values were set to -16 , the next negative integer of the most negative log2-transformed value of the dataset. We aim to visualize the data in a way

that allows data extraction in a spatial context. The computational basis to achieve this objective is PCA-based 3D data projection.

MATERIALS

REAGENTS

- Example dataset (see Reagent setup)
- Single-cell gene expression data from a hollow, spherically-shaped organ or tissue. Data may be generated using qRT-PCR-based³¹ or RNA-Seq-based²⁸ approaches. Expression values should be organized in matrix format such that rows correspond to cells and columns correspond to genes.

EQUIPMENT

Hardware

- A desktop computer or laptop. Please note that this protocol can be implemented on Windows as well as on UNIX based operating systems (e.g., Mac OS X, Linux).

Software

- One of these two alternatives: The MATLAB software, commercially available from <http://www.mathworks.com/> or the free R statistical computing environment, downloadable from <http://www.r-project.org/>
- Optional: Microsoft Excel software, commercially available from <http://office.microsoft.com/en-us/excel/>

REAGENT SETUP

Input file formats—Multiple file formats can be used to import the data into the MATLAB or R platforms. The raw data format created by the Fluidigm Software is a comma-separated value file (CSV), which can be opened and processed with any text editor, as well as Microsoft Excel. Processed RNA-Seq data is commonly available in .TXT or .CSV file format.

Downloading example datasets—Supplementary Table S2 of Durruthy-Durruthy, R. *et al.*⁶ contains the 382-cell (otocyst and neuroblast) data table in a pre-processed and summarized format in an Excel file (mmc2.xlsx, available from the journal's webserver)⁶. It is not the intent of this protocol to discuss the methods (e.g., measures to assess quality of single cell data, normalization) for obtaining ready-to-use data from recorded raw data, which should be determined at the researcher's discretion. The example RNA-Seq data⁷ can be downloaded from the Gene Expression Omnibus (GEO) database under accession number GSE59130. Alternatively, both files are included as Supplementary Data in this protocol. Within this zip file are Otocyst_data.csv and renalvesicle_data.csv, which include only the relevant data, comprised of expression values of the 267 otocyst cells and 57 renal vesicle cells, respectively. Download and save either of these files to the working directory.

EQUIPMENT SETUP

MATLAB software installation—Download and install the latest version of MATLAB from <http://www.mathworks.com/products/matlab/>. Please refer to the product manual for information regarding the download and installation procedure. Another useful resource for MATLAB can be found at: http://www.mathworks.com/videos/getting-started-with-matlab-68985.html?s_tid=main_tutorial_ML_rp. Please verify that the required MATLAB toolboxes are installed (Table 1).

MATLAB function files—Download and unpack the file Supplementary Software 1. Save all function files in the MATLAB working directory (the same directory where the data files are saved). Each program file is a function that is used throughout the protocol. For additional information about each file please type 'help program file' at the MATLAB command line.

R software Installation—Download and install the latest release version of R from <http://cran.r-project.org/>. Consult the R installation and administration manual if necessary. A useful reference for R commands can be found at <http://cran.r-project.org/doc/contrib/Short-refcard.pdf>. On Unix-based operating systems (e.g., Linux and OSX), the R package that plots data in three dimensions requires that X11 be installed. Additional required packages are specified in the Procedure section and in Table 1.

R script files—Download and unpack the file Supplementary Software 2. Save all script files in the R working directory (the same directory where the data files are saved). For additional information about each file please type '?program file' at the command.

PROCEDURE

CRITICAL: Below is described the application of the protocol using MATLAB. Two alternatives are given: one that makes use of the otocyst dataset (Step 2 and Step 3, Options A), the other that makes use of the renal vesicle dataset (Step 2 and Step 3, Options B). An analogous step-by-step procedure that uses the software environment R is provided in Supplementary Methods. Users may choose either one based on their preferences.

CRITICAL: In the following, MATLAB commands will be preceded by a '>>' sign, which is the typical prompt in MATLAB, not part of the executed command. Each command has to be executed in the sequence specified in the protocol. A semicolon at the end of a command tells MATLAB not to display any output from the command. An individual command preceded by a '>>' sign may stretch continuously over multiple lines.

Importing data into MATLAB workspace **TIMING 10 min**

1. Open the MATLAB software and choose a 'working directory' at the left window area by clicking 'Browse for folder' at the toolbar. Choose the working directory (shown in the 'Current Folder' window) where you saved the data and function files. Alternatively, you can use the cd command to set the working directory. E.g., if the working directory path is C:/MyWorkingDirectory (e.g., on Windows), execute from the command prompt:

```
>> cd C:/MyWorkingDirectory
```

CRITICAL STEP: The 'working directory' is important for accessing and loading files when they are called at the prompt in the Command Window. If uncertain, type pwd at the prompt to retrieve the path to the working directory.

TROUBLESHOOTING

2. Import data matrix into MATLAB, according to option A for the otocyst-based dataset, or option B for the renal vesicle-based dataset (please note that, in order to familiarize themselves with the Procedure, readers may decide to implement both options).

A. Importing data from otocyst-derived cells

- i. (OPTIONAL) If using Table S2 from Durruthy-Durruthy *et al.*⁶, execute the following commands:

```
>> [~,~,alldata] = xlsread('mmc2.xlsx',5,'','basic');
>> data = cell2mat(alldata(12:393,8:103));
>> genes = alldata(11,8:103);
>> cells = alldata(12:393,1);
>> clusterLabels = alldata(12:393,4);
>> otocystCluster = ismember(clusterLabels,
    {'B2';'B4';'B5';'B6'});
>> data = data(otocystCluster,:);
>> cells = cells(otocystCluster,:);
```

These commands load the expression values of the 382 cells (= observations) over the 96 genes (= variables) as a matrix into the data variable, the gene names into genes variable and uses the bi-cluster labels (B2, B4-B6) to extract only the 267 cells related to the otocyst. For compatibility with non-Windows operating systems, the Excel sheet is loaded in basic mode.

- ii. (OPTIONAL) Alternatively, load the otic data matrix of expression values from only the 267 otocyst cells from a CSV file provided in the Supplementary Data by executing the following commands:

```
>> dataFileName = 'otocyst_data.csv';
>> data = csvread(dataFileName,1);
>> cells = data(:,1);
>> data = data(:,2:end);
```

```

>> fileID = fopen(dataFileName);
>> line = fgetl(fileID);
>> genes = regexp(line,',','split');
>> genes = genes(2:end);
>> fclose(fileID);

```

Here, the `csvread` function is used to read the numeric data. For the gene names, open the file using `fopen` and retrieve the first line using `fgetl`. Lastly, split the comma-delimited line to gene names and close the file (`fclose`). After a successful import, the variables may be found in the 'Workspace' window. The numeric matrix data contains all expression values in array format. The cell array `genes` lists all gene symbols in a vector format.

- iii. (OPTIONAL) Alternatively, users could instead use the following function that compiles all listed commands to load all expression values and gene names in one step:

```

>> [data,genes] = LoadData(dataFileName);

```

Here `dataFileName` corresponds to either 'mmc2.xlsx' or 'otocyst_data.csv'.

TROUBLESHOOTING

B. Importing data from renal vesicle-derived cells

- i. (OPTIONAL) Load the expression values from Brunskill *et al.*⁷ from the CSV file provided in the Supplementary Data implementing the following commands:

```

>> dataFileName = 'renalvesicle_data.csv';
>> data = csvread(dataFileName,1);
>> cells = data(:,1);
>> data = data(:,2:end);
>> fileID = fopen(dataFileName);
>> line = fgetl(fileID);
>> genes = regexp(line,',','split');
>> genes = genes(2:end);
>> fclose(fileID);

```

- ii. (OPTIONAL) Alternatively, users could instead use the following function that compiles all listed commands to load all expression values and gene names in one step:

```
>> [data,genes] = LoadData('renalvesicle_data.csv');
```

TROUBLESHOOTING

Identifying anchor gene–correlated markers for principal component analysis **TIMING 5 min**

CRITICAL: Accurate approximation of spatial relationships between single cells in the three-dimensional space requires the inclusion of features that demonstrate anatomical asymmetries in expression that are axially manifested. Conclusive representation of the tissue or organ involves, prior to PCA, identifying gene markers with spatially associated information and dismissing genes with cell-to-cell expression level variability that does not correlate with cellular position. The following Procedure step 3 describes the selection of marker genes with known asymmetric expression distribution that selectively and broadly label cells of the particular domain one seeks to model (e.g., *Oc90* for the dorsal otocyst domain). We refer to these markers as 'anchor genes' (see Experimental Design section of Introduction for further information). In the script file DetermineConclusiveAG.m within the Supplementary Software 1 we list the conceptual steps of this strategy in detail in the programmer comments.

3. Use the following commands to assign anchor genes for each axis that one seeks to compute and remove control genes if necessary (see Experimental Design section of the Introduction for further information on selecting anchor genes and control genes). Here, genes with putatively axially defined expression domains are provided for up to two anatomical axes (Axis1, Axis2). Each of the axes divides the spherical model in two hemispheres (Side1, Side2). Assign and remove genes according to option A if working with the otocyst sample, or according to option B if working with the renal vesicle sample.

A. Assigning anchor genes for the otocyst sample

- i. Type in the command line the following commands:

```
>> anchorGenesAxis1Side1 = {'Oc90'};
>> anchorGenesAxis1Side2 = {};
>> anchorGenesAxis2Side1 = {'Gbx2'};
>> anchorGenesAxis2Side2 = {};
>> genes2Remove = {'Actb'; 'Gapdh'; 'Egfp'; 'Pax6'};
>> gene2show = 'Oc90';
>> howManyDomains = 8;
```

These commands will assign *Oc90* as an anchor gene that characterizes the first axis (here dorsal-ventral) and *Gbx2* as an anchor gene that defines the second axis (here lateral-medial). Both genes are chosen based on their well-documented expression domains in the literature. In this example, *Oc90* is selectively expressed in cells on the dorsal

half of the mouse otocyst²¹ (Figure 1b), and *Gbx2* labels preferentially cells on the medial side of the tissue²². Through the above commands, *Actb*, *Gapdh*, and *Egfp* are removed as positive control markers. All three markers are not otic-specific and expression of these markers was evaluated for data quality measures. *Pax6* is removed in this example as a negative marker as its expression served to identify potential nonotic cells and was only expressed in one cell. Please note that the script also contains a function that automatically removes genes with uniform expression across all cells, such as genes that are not expressed in any of the cells. Expression values of *Oc90* will be initially used to color-code the cells arranged within a sphere that models the shape of the organ or tissue the cells are from. The gene whose expression values are to be shown can be changed as desired by specifying the gene in the `gene2show` command line, above. `howManyDomains` specifies the number of domains requested by the user. Possible values are 2 for hemispheres, 4 for quadrants and 8 for octants and should be selected according to the availability of anchor genes. Providing anchor genes for only one axis (Axis 1) allows for the distinction of two domains only (parameter value 2). Providing anchor genes for two axes (Axis 1 and Axis 2) enables the determination of more constricted expression territories (parameter value 4 or 8) (Figure 6). Here, in the case of the otocyst, eight domains are chosen for the following reasons: Segmenting the model into eight parts enables the user to define more refined domains such that all three main anatomical axes are considered when referring to a particular territory (e.g., ventral-lateral-anterior). A number of different candidate markers are available that have described expression domains and cover all three axes. Subsequent quantitation of expression levels may be performed on the basis of these defined domains and aid in evaluating the overall accuracy of the model.

TROUBLESHOOTING

B. Assigning anchor genes for the renal vesicle sample

i. Type in the command line the following commands.

```
>> anchorGenesAxis1Side1 = {'Pou3f3'; 'Sox9';
    'Dkk1'; 'Greb1'; 'Dll1'; 'Papss2'; 'Lhx1'; 'Pcsk9';
    'Bmp2'};

>> anchorGenesAxis1Side2 = {'Cdh6'; 'Wt1';
    'Tmem100'};
```

```

>> anchorGenesAxis2Side1 = {};
>> anchorGenesAxis2Side2 = {};
>> genes2Remove = {};
>> gene2show = 'Dll1';
>> howManyDomains = 2;

```

These commands will assign all genes listed in the first two commands as anchor genes that characterize the first axis (here proximal-distal). No additional genes are listed to describe a second axis or to be removed in this example. Expression values of *Dll1* will serve initially as data to color-code the cells arranged within a sphere that models the shape of the organ or tissue the cells are from and can optionally be changed.

TROUBLESHOOTING

4. As data and parameter integrity are crucial for the success of the protocol, run the following command to verify the completeness of both and review your data if necessary:

```

>> [data,howManyDomains] = CheckIntegrity(data, genes,
    anchorGenesAxis1Side1, anchorGenesAxis1Side2,
    anchorGenesAxis2Side1, anchorGenesAxis2Side2,
    gene2show,genes2Remove, howManyDomains);`

```

This command will initiate warnings if data or parameter integrity is compromised. It inspects the data for missing expression values, and validates that the parameters are of correct data type so that calculations can be computed without logical errors. It also verifies that the number of requested domains is compatible with the number of input axes. TROUBLESHOOTING

5. Discriminate between conclusive and inconclusive anchor genes. Anchor genes whose level of expression highly correlates with that of other anchor genes of the same axis orientation (dorsal-ventral and lateral-medial, respectively for otocyst; proximal-distal for renal vesicle) are included for further analysis. Here, Pearson's correlation coefficients are used as a measure of similarity and markers with associated coefficient values above 0.25 (positive correlation) or below -0.25 (negative correlation), respectively, are considered. Further filtering separates non-conclusive anchor genes from conclusive anchor genes. In particular only anchor genes whose expression profiles correlate with at least 50% of all other anchor genes are considered as conclusive (see DetermineConclusiveAG.m in Supplementary Software 1). To display all conclusive and inconclusive anchor genes type the following commands in the command line.


```
>> [conclusiveAGAxis1Side1, conclusiveAGAxis1Side2] =
    DetermineConclusiveAG(data, genes, anchorGenesAxis1Side1,
    anchorGenesAxis1Side2,1)

>> [conclusiveAGAxis2Side1, conclusiveAGAxis2Side2] =
    DetermineConclusiveAG(data, genes, anchorGenesAxis2Side1,
    anchorGenesAxis2Side2,2)
```

Implementing these commands will return conclusive anchor genes for each axis. The information is stored in cell arrays and can be found in the 'Workspace' window. If anchor genes are excluded from the analysis because of low correlation, a warning will appear with a list of these genes.

6. Identify all other genes that correlate with all conclusive anchor genes (known as anchor gene-correlated genes, Figure 3a). Here, all genes in the dataset whose expression profiles correlate with expression profiles of at least 50% of conclusive anchor genes are identified. Expression values of these genes will be applied for subsequent principal component analysis. Type the following command in the command line.

```
>> spatialGenes = FilterIrrelevantGenes (data, genes, genes2Remove,
    conclusiveAGAxis1Side1, conclusiveAGAxis1Side2,
    conclusiveAGAxis2Side1, conclusiveAGAxis2Side2, 1);
```

Implementing this command will withdraw: control genes that are requested to be removed by the user (genes2Remove); genes that show equal expression across all cells (or no expression at all); and all genes whose expression is not correlated with any of the conclusive anchor genes ($-0.25 < \text{Pearson's correlation coefficient} < 0.25$). The final list of genes included for downstream analysis is returned in the spatialGenes variable.

PCA and examination of variance distribution across first principal components **TIMING 10 min**

CRITICAL: This subsection of the Procedure is Optional. It can be implemented to examine the data variance that is retained in the first few principal components (Figure 3b). Cases in which the variance in the first three components is not proportionally higher than in the subsequent components, may result in less accurate representations of the tissue or organ. This is crucial as data projection onto a lower-dimensional subspace, which is defined by principal components, is always accompanied with some degree of information loss. A relatively large total variance value that is captured by the first three components in relation to the following components suggests that information loss is attenuated which in turn improves pattern recognition of the data. This section also enables the visual inspection of the data when projected onto the first three components. Identification of subgroups based on how the data spreads in this subspace may lead to non-homogenous distributions of cells when projected onto a spherical model. In both cases (low variance in first three components, subgroup recognition) an alternative selection of anchor genes is advised.

7. Perform PCA (for more information on the pca toolbox, refer to the file information or call the help file by typing `help pca' at the prompt in the command window) implementing the following commands.

```
>> filteredData = data(:, ismember(genes, spatialGenes));
>> [~,scores,~,~,explained] = pca(filteredData);
```

This command calculates the relevant parameters required to plot the data onto the newly transformed coordinate system. The following output variables are computed and listed in the workspace window.

scores = 267×29 numeric matrix that presents the transformed data (otocyst) = 57×731 numeric matrix that presents the transformed data (renal vesicle)

explained = numeric vector that lists the percentage of the original total variance that is explained by each of the principal components.

8. Use the pareto function to plot the variance that is retained. For this purpose, type the following commands at the prompt.

```
>> figure;
>> pareto(explained);
>> xlabel('Principal Component');
>> ylabel('Variance Retained [%]');
```

Implementation of these commands will plot the variance that is retained in each of the first ten principal components in descending order (bars) plus the accumulated total variance across PCs (line) (Figure 3b).

TROUBLESHOOTING

9. Project data onto the first three components (Figure 4). For this purpose, at the prompt type the following commands (for more information of the scatter3 toolbox, please consult the relevant manual or type `help scatter3' at the prompt sign).

```
>> figure();
>> scatter3(scores(:,1),scores(:,2),scores(:,3));
>> xlabel('PC 1');
>> ylabel('PC 2');
>> zlabel('PC 3');
```

An external window will open and plot the data projection. The researcher can rotate the object unrestrictedly around all three axes using the Camera Toolbar (activate under `View' in the toolbar). Here, the position of each data point (individual cell) is purely based on the expression values of the data matrix as the three axes represent contributions of all genes with putatively spatial information assayed. TROUBLESHOOTING

Establishing boundaries and projecting the data onto 3D space TIMING 10 min

CRITICAL: Boundary formation between groups of cells is a crucial event in development, as it enables cells with distinct functions to be kept physically separated from each other. In addition, the shape of a boundary and its orientation with respect to the body axes can influence important patterning events of an organ as development progresses and control its morphogenic roadmap. Here, boundaries are computed on the basis of co-expression of all conclusive anchor genes for each domain. Specific details of how the borders between neighboring domains are established can be found in the programmer comments in the script file `ExtractCellsForAxis.m` within the Supplementary Software 1.

10. For each axis, identify the cells that express conclusive anchor genes on one side and do not express the conclusive anchor genes that are typically expressed on the opposite side. The numbers of cells that express and do not express anchor genes is a good measure whether overall distribution of cells lies within the users expectations. For this purpose, obtain cell number assignment by typing the following commands in the command line.

```
>> cellsForAxis1 = ExtractCellsForAxis(data, genes,
    conclusiveAGAxis1Side1, conclusiveAGAxis1Side2);

>> cellsForAxis2 = ExtractCellsForAxis(data, genes,
    conclusiveAGAxis2Side1, conclusiveAGAxis2Side2);
```

The returned matrices `cellsForAxis1` and `cellsForAxis2` include the list of cells that are considered for centroid computation and boundary formation. Here, in particular cells that co-express more than 50% of all conclusive anchor genes for one side of the axis, and at the same time express less than 50% of all conclusive anchor genes of the other side of the same axis are included. For each side, the centroid's coordinates in the three-dimensional system are based on the mean position of all conclusive anchor genes that fulfill above criteria. **CRITICAL:** The command for cell number assignment must be implemented separately for both axes, as shown above. If second-axis-anchor-genes are not provided, `cellsForAxis2` will be an empty matrix.

11. The final number of cells per axis that qualify to be included for boundary formation computation may vary depending on the initial anchor genes selection (Step 3). Execute the following command to display the final number of cells for the first axis.

```
>> display([num2str(sum(cellsForAxis1(:,1) | cellsForAxis1(:,
    2))), '/', num2str(size(cellsForAxis1, 1)), ' cells are expressing the
    following anchor genes:', strjoin([conclusiveAGAxis1Side1;
    conclusiveAGAxis1Side2], ', ')]);
```

TROUBLESHOOTING

12. (OPTIONAL) Similarly, if anchor genes are supplied for the second axis, enter the following command to display the final number of cells expressing the anchor genes for the second axis.

```
>> display([num2str(sum(cellsForAxis2(:,1) | cellsForAxis2(:,
2))),',',num2str(size(cellsForAxis2, 1)),', cells are expressing the
following anchor genes:', strjoin([conclusiveAGAxis2Side1;
conclusiveAGAxis2Side2'],' ')]);
```

13. To project data on three dimensions in spherical form and color-code cells based on expression levels for gene of interest (Figure 4), type the following command in the command line, replacing the variable `gene2show` with any gene name that one seeks to plot (i.e., `Dlx5`):

```
>> cellAssignments = PlotCellsIn3D(data, genes, spatialGenes,
gene2show, cellsForAxis1, cellsForAxis2, howManyDomains);
```

Following the command, an external window will open and plot the data projection of the 267 cells (otocyst) or 57 cells (renal vesicle) onto the hollow unit sphere. Here the position of each data point (individual cell) is based on the expression values of the data matrix as the three axes represent contributions of all genes assayed. The researcher can rotate the object unrestrictedly around all three axes using the Camera Toolbar (activate under `'View'` in the toolbar), or change the colormap by either right clicking on the colorbar (on the right side of the figure) and choosing other available colormaps from the Standard Colormaps menu (Windows only) or opening the Colormap Editor in Menu `"Edit" -> "Colormap..."`.

CRITICAL STEP: Depending on the number of axes determined previously (step 3) the sphere is partitioned into two, four or eight domains. The `PlotCellsIn3D` function outputs the assignment of cells to domains (hemispheres, quadrants or octants) in the variable `cellAssignments`, which is a vector of length equal to the number of cells containing the numeric assignments of cells to the different hemispheres (values are 1 or 2), quadrants (i.e., Dorsal-Ventral and Lateral-Medial axes), or octants on for instance the Dorsal-Ventral, Lateral-Medial and Anterior-Posterior axes. The assignments can be inferred from Figure 6. For brevity, we do not specify the entire code of this function. Its main functional principles are documented in detail in the file `PlotCellsIn3D.m` within Supplementary Software 1. We also postulate that all three major body axes are aligned perpendicular to each other. This will facilitate visualization, characterization and quantification of the mathematical projection.

Quantitation of the data **TIMING 5 min**

14. Use the following command to return for each domain the relative percentage of cells that express a gene and the mean transcript levels of genes (Figure 5), replacing the variable `gene2show` by any gene name that one seeks to analyze; i.e., `Dlx5`):

```
>> [numOfCellsExpressingPerDomain, meanExpPerDomain,
stdExpPerDomain] = PlotGeneExpressionOverSphere(data, genes,
gene2show, cellAssignments,howManyDomains);
```

This will return a graphical representation of the quantitation in histogram format. Outliers are shown as red '+' signs. It will also return the following output variables in column vector format containing domain-associated quantitative values:

numOfCellsExpressingPerDomain

= Percentage of cells per domain that expresses a particular gene.

meanExpPerDomain

= Mean expression levels of cells per domain. The average expression levels are calculated based on all cells that express the gene.

stdExpPerDomain

= Standard deviation of mean expression of cell per domain.

15. (OPTIONAL) If the user is familiar with Steps 1–14 of the protocol and has determined appropriate genes to be included and excluded, as well as appropriate anchor genes, he or she might find it useful to execute all commands required for generating a spherical projection at once. This is done in a one-line command where the user can conveniently adjust input parameters and rapidly map expression distributions of various genes onto the sphere model. This is particularly useful in situations where many projections need to be generated or when the user quickly seeks to compare models generated with different parameters. This command only generates spherical projections and does not provide details of data integrity (Step 4), PCA variance (Step 8), and quantitation (Step 14). At the command line type the following but replace the respective descriptors with specific values (i.e. dataFileName = 'mmc2.xlsx'; see the example below the command for details):

```
>> [data, genes, cellAssignments] = RunPlotCells3D(dataFileName,
    anchorGenesAxis1Side1, anchorGenesAxis1Side2,
    anchorGenesAxis2Side1, anchorGenesAxis2Side2, gene2show,
    genes2Remove, filterUncorrGenes, howManyDomains)
```

Example (using using Table S2 from Durruthy-Durruthy *et al.*⁶, 'mmc2.xlsx'):

```
>> [data, genes, cellAssignments] = RunPlotCells3D('mmc2.xlsx',
    {'Oc90'}, {}, {'Gbx2'}, {}, 'Oc90', {'Actb'; 'Gapdh'; 'Egfp'}, 1, 8);
```

TIMING

Steps 1 – 2: Importing data into MATLAB workspace: 10 min.

Steps 3 – 6: Determine anchor gene correlated markers for principal component analysis: 5 min.

Steps 7 – 9 (Optional): Perform principal component analysis and examine variance distribution across first principal components: 10 min.

Steps 10 – 13: Establish boundaries and project the data onto 3D space: 10 min.

Step 14: Quantitation of the data: 5 min.

Step 15: One-line command to plot expression data in 3D space: 2 min.

ANTICIPATED RESULTS

Principal component analysis is a mathematical procedure that aims to reduce the number of informative variables into a smaller set. As a measure of biological variability, the proportion of the total variance retained within each principle component is maximized and PCs are ranked in descending order (PC1 with highest variance, PC2 with second highest variance, etc.). In our setting, the first three components comprise 49% (otocyst) and 28% (renal vesicle) of the total variance when PCA is performed with all genes whose expression differences across cells are presumably based on spatial information (Figure 3a,b). We cannot provide a definitive figure that would enable the researcher to estimate whether the subspace that is spanned by the first three components retains 'enough' data variability. Additional studies in the future that will produce single-cell gene expression data of spherical organs will likely contribute to a more determined assessment of this measure. Therefore, it is up to the researcher to decide whether a sufficient portion of the total variance is accounted by the first three components. For the two example data sets used in the protocol, we quantitated the expression distribution of various genes and found that their distribution aligned well with previously defined expression domains in the developing organ as described in the literature for the ear⁶ and renal vesicle⁷ (Figure 5). Based on these findings, we concluded that the three-component-subspace generated by PCA captures an adequate amount of data variability. If the first three PCs account for an insufficient portion of the total variance, it may exacerbate the difficulty of recognizing patterns and of performing quantitative downstream evaluations. To overcome such shortcomings, a revised gene collection may be needed.

The graphical output obtained at the end of the Procedure should architecturally resemble the crude geometry of a shell-like spherical object (Fig. 4). Morphologically more complex organs and tissues will demand alternative mathematical approaches, as outlined in the Limitations section.

Simple quantification methods based on different segmentation measures (e.g., other than Apical-Posterior, Dorsal-Ventral, Lateral-Medial, Proximal-Distal distinction) of the object can also be applied to support different approaches to characterize gene expression patterns.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGEMENTS

We thank all members of the Heller laboratory for comments on the manuscript. This work was supported by NIH grants DC006167 and DC012250 to S.H., by P30 core support (DC010363), by the Stanford Initiative to Cure Hearing Loss, and in part by FP7-Health-2013-Innovation a cooperative grant by the European Commission.

REFERENCES

1. Guo G, et al. Resolution of cell fate decisions revealed by single-cell gene expression analysis from zygote to blastocyst. *Dev Cell*. 2010; 18:675–685. doi:10.1016/j.devcel.2010.02.012. [PubMed: 20412781]
2. Dalerba P, et al. Single-cell dissection of transcriptional heterogeneity in human colon tumors. *Nat Biotechnol*. 2011; 29:1120–1127. doi:10.1038/nbt.2038. [PubMed: 22081019]
3. Moignard V, et al. Characterization of transcriptional networks in blood stem and progenitor cells using high-throughput single-cell gene expression analysis. *Nat Cell Biol*. 2013; 15:363–372. doi:10.1038/ncb2709. [PubMed: 23524953]
4. Guo G, et al. Mapping cellular hierarchy by single-cell analysis of the cell surface repertoire. *Cell Stem Cell*. 2013; 13:492–505. doi:10.1016/j.stem.2013.07.017. [PubMed: 24035353]
5. Narsinh KH, et al. Single cell transcriptional profiling reveals heterogeneity of human induced pluripotent stem cells. *J Clin Invest*. 2011; 121:1217–1221. doi:10.1172/JCI44635. [PubMed: 21317531]
6. Durruthy-Durruthy R, et al. Reconstruction of the mouse otocyst and early neuroblast lineage at single-cell resolution. *Cell*. 2014; 157:964–978. doi:10.1016/j.cell.2014.03.036. [PubMed: 24768691]
7. Brunskill EW, et al. Single cell dissection of early kidney development: multilineage priming. *Development*. 2014; 141:3093–3101. doi:10.1242/dev.110601. [PubMed: 25053437]
8. Elowitz MB, Levine AJ, Siggia ED, Swain PS. Stochastic gene expression in a single cell. *Science*. 2002; 297:1183–1186. doi:10.1126/science.1070919. [PubMed: 12183631]
9. Yeung KY, Ruzzo WL. Principal component analysis for clustering gene expression data. *Bioinformatics*. 2001; 17:763–774. [PubMed: 11590094]
10. Ringner M. What is principal component analysis? *Nat Biotechnol*. 2008; 26:303–304. doi:10.1038/nbt0308-303. [PubMed: 18327243]
11. Neves SR, et al. Cell shape and negative links in regulatory motifs together control spatial information flow in signaling networks. *Cell*. 2008; 133:666–680. doi:10.1016/j.cell.2008.04.025. [PubMed: 18485874]
12. Fuhrmann S. Eye morphogenesis and patterning of the optic vesicle. *Curr Top Dev Biol*. 2010; 93:61–84. doi:10.1016/B978-0-12-385044-7.00003-5. [PubMed: 20959163]
13. Brewster SF. The development and differentiation of human seminal vesicles. *J Anat*. 1985; 143:45–55. [PubMed: 3870731]
14. Essner JJ, Amack JD, Nyholm MK, Harris EB, Yost HJ. Kupffer's vesicle is a ciliated organ of asymmetry in the zebrafish embryo that initiates left-right development of the brain, heart and gut. *Development*. 2005; 132:1247–1260. doi:10.1242/dev.01663. [PubMed: 15716348]
15. Weninger WJ, Mohun T. Phenotyping transgenic embryos: a rapid 3-D screening method based on episcopic fluorescence image capturing. *Nat Genet*. 2002; 30:59–65. doi:10.1038/ng785. [PubMed: 11743576]
16. Schena M, Shalon D, Davis RW, Brown PO. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*. 1995; 270:467–470. [PubMed: 7569999]
17. Fernandez R, et al. Imaging plant growth in 4D: robust tissue reconstruction and lineaging at cell resolution. *Nat Methods*. 2010; 7:547–553. doi:10.1038/nmeth.1472. [PubMed: 20543845]
18. Lee JH, et al. Highly multiplexed subcellular RNA sequencing in situ. *Science*. 2014; 343:1360–1363. doi:10.1126/science.1250212. [PubMed: 24578530]
19. Jolliffe, IT. Principal component analysis. 2nd edn. Springer; 2002.
20. Scholz M, Kaplan F, Guy CL, Kopka J, Selbig J. Non-linear PCA: a missing data approach. *Bioinformatics*. 2005; 21:3887–3895. doi:10.1093/bioinformatics/bti634. [PubMed: 16109748]
21. Verpy E, Leibovici M, Petit C. Characterization of otoconin-95, the major protein of murine otoconia, provides insights into the formation of these inner ear biominerals. *Proc Natl Acad Sci U S A*. 1999; 96:529–534. [PubMed: 9892667]

22. Lin Z, Cantos R, Patente M, Wu DK. Gbx2 is required for the morphogenesis of the mouse inner ear: a downstream candidate of hindbrain signaling. *Development*. 2005; 132:2309–2318. doi: 10.1242/dev.01804. [PubMed: 15829521]
23. Hurd EA, Poucher HK, Cheng K, Raphael Y, Martin DM. The ATP-dependent chromatin remodeling enzyme CHD7 regulates pro-neural gene expression and neurogenesis in the inner ear. *Development*. 2010; 137:3139–3150. doi:137/18/3139 [pii] 10.1242/dev.047894. [PubMed: 20736290]
24. Picelli S, et al. Full-length RNA-seq from single cells using Smart-seq2. *Nat Protoc*. 2014; 9:171–181. doi:10.1038/nprot.2014.006. [PubMed: 24385147]
25. Citri A, Pang ZP, Sudhof TC, Wernig M, Malenka RC. Comprehensive qPCR profiling of gene expression in single neuronal cells. *Nat Protoc*. 2012; 7:118–127. doi:10.1038/nprot.2011.430. [PubMed: 22193304]
26. Schmitz B, et al. Magnetic activated cell sorting (MACS)--a new immunomagnetic method for megakaryocytic cell isolation: comparison of different separation techniques. *European journal of haematology*. 1994; 52:267–275. [PubMed: 8020626]
27. Espina V, et al. Laser-capture microdissection. *Nat Protoc*. 2006; 1:586–603. doi:10.1038/nprot.2006.85. [PubMed: 17406286]
28. Tang F, et al. mRNA-Seq whole-transcriptome analysis of a single cell. *Nat Methods*. 2009; 6:377–382. doi:nmeth.1315 [pii] 10.1038/nmeth.1315. [PubMed: 19349980]
29. Trapnell C, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nat Biotechnol*. 2014; 32:381–386. doi:10.1038/nbt.2859. [PubMed: 24658644]
30. Pollen AA, et al. Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat Biotechnol*. 2014 doi:10.1038/nbt.2967.
31. Sanchez-Freire V, Ebert AD, Kalisky T, Quake SR, Wu JC. Microfluidic single-cell real-time PCR for comparative analysis of gene expression patterns. *Nat Protoc*. 2012; 7:829–838. doi:10.1038/nprot.2012.021. [PubMed: 22481529]

Figure 1. Comparison of Different Technologies to Study Gene Expression Levels

(a) Three different widely used technologies with their main advantage indicated. Each grey-colored circle lists exemplary approaches for each technology. The intersect of all three major advantages (High-Throughput, Intact Morphology, Cell-to-Cell Resolution) is indicated in red and highlights the significance of the herein described approach. **(b)** Schematically shown is how the Protocol translates gene expression information from the organ to a computer-generated model. `Ex vivo' shows an *in situ* hybridization for *Oc90* mRNA on a section of the mouse otocyst visible as blue coloring in the dorsal section of the organ. `In silico' shows the expression pattern of *Oc90* mRNA in the reconstructed 3D-model where expression of the gene is indicated by colored circles that represent individual cells. Color code for gene expression levels: red=high, orange/yellow=medium, green=low, grey=not detectable.

Figure 2. Overview of the Protocol Workflow
Modules can be completed by various alternative approaches ('Methodology'), which can be customized to the researcher's scientific question. Yellow-labeled box indicates a step of critical importance. Modules colored in orange are discussed in this present protocol and relevant Procedure Steps are indicated.

Figure 3. Anchor Gene Correlation and Principal Component Analysis

(a) Schematic overview of compiling anchor gene (AG)-correlated genes for the otocyst (blue) and the renal vesicle (orange). Shown is a heatmap in diagram format that represents the high-dimensional numerical data where rows equal single cells and columns represent genes (variables). The number of variables that is used for PCA and subsequent data projection is reduced and consists only of genes that correlate with pre-defined spatially informative anchor genes. This panel relates to Steps 3–6 of the Procedure. (b) Variance distribution across the first three principal components (PCs) for otocyst (top panel) and renal vesicle (bottom panel). Side-by-side comparison demonstrates the effects of reducing the number of variables: In the case of the otocyst, the variance captured by the first three PCs is almost half of the total variability of the dataset captured when considering only anchor gene–correlated genes for PCA, compared to only 34% of the total accumulative variance when the PCA was conducted with all genes. In the case of the renal vesicle, this difference is even more pronounced. If PCA is performed with all genes, the first three PCs only account for 13% of total variance compared to 28% if anchor gene-correlated genes are utilized. This panel relates to Steps 3–8 of the Procedure.

Figure 4. Data projection onto the first three PCs

From left to right is reported how cells are projected onto the first three PCs (left), how their position is adjusted by ordinate origin distance normalization (middle) and how cells are color-coded based on expression levels of one example anchor gene (right). For otocyst (top, blue) and renal vesicle (bottom, orange) projections are shown if all genes are used for PC analysis or only anchor gene correlated genes are considered. This panel relates to Steps 9–13 of the Procedure.

Figure 5. Visualization and quantitation of otocyst and renal vesicle models

Cells are color-coded in the 3D projections shown to illustrate expression levels of example genes (Oc90, Gbx2, Pou3f3, Wt1) with reported axially-confined expression domains. The otocyst model (top) is partitioned into eight octants as a result of dorsal/ventral, medial/lateral, and anterior/posterior distinction. The renal vesicle model (bottom) is divided into two hemispheres that indicate proximal/distal domains. For each representative gene, domain-associated cell percentage is shown that expresses a gene above detection threshold (top graph) and mean expression for the gene is indicated (bottom graph). Whiskers of box plots present lowest value still within 1.5 interquartile range (IQR) and highest value still within 1.5 IQR. Median = vertical line. Mean = plus sign. Only data points with true expression values (e.g., data points that are not empty) are considered for average expression calculations. This panel relates to Step 14 of the Procedure.

Figure 6. Sphere partitioning
Depending on the number of domains specified by the user, the sphere is partitioned into two hemispheres, four quadrants or eight octants. For each of the three axes (Axis1-Axis3) two different sides are shown (Side 1/Side 2) for which anchor genes can be user-defined in step 3 of the procedure. Shown are also the assignment numbers (‘Assignment’) per domain that relate to the quantitative analysis output of the procedure in Steps 10–14 (e.g., when 8 domains are designated by the user (‘Octants’), the assignment numbers 1–4 correspond to the domain that is characterized by expression of anchor gene(s) of Side 1 on the first axis (S1 = Side 1)).

Table 1

Toolboxes and packages used

Toolboxes (MATLAB) and Packages (R)	Description	Additional Information
MATLAB		
Statistics Toolbox	Provides statistical algorithms to analyze data	http://www.mathworks.com/products/statistics/
Symbolic Toolbox	Provides functions for solving math expressions	http://www.mathworks.com/products/symbolic/
R		
rgl Package	Provides medium for three dimensional visualization	http://cran.rproject.org/web/packages/rgl/rgl.pdf
colorRamps Package	Provides medium to build gradient color maps	http://cran.rproject.org/web/packages/colorRamps/colorRamps.pdf
XLConnect Package	Provides medium to manipulate Microsoft Excel files from R	http://cran.rproject.org/web/packages/XLConnect/XLConnect.pdf
Functions (selection)	Description	Additional Information
MATLAB		
csvread	Read comma-separated value files	http://www.mathworks.com/help/matlab/ref/csvread.html
pea	Principal component analysis of raw data	http://www.mathworks.com/help/stats/pca.html
scatter3	Plots scatter plot in three dimensions	http://www.mathworks.com/help/matlab/ref/scatter3.html
xlsread	Read Microsoft Excel spreadsheet files	http://www.mathworks.com/help/matlab/ref/xlsread.html
R		
princomp	Principal component analysis of raw data	http://stat.ethz.ch/R-manual/R-patched/library/stats/html/princomp.html
read.csv	Read comma-separated value files	http://cran.rproject.org/doc/manuals/r-release/R-data.pdf
readWorksheetFromFile	Read Microsoft Excel spreadsheet files (XLConnect Package)	http://cran.rproject.org/web/packages/XLConnect/XLConnect.pdf

Table 2

Troubleshooting

step	Problem	Possible Reason	Solution
1	The working directory is not set correctly	The path is not correct	Make sure the path is the working directory is valid
2A iii, 2B ii	Cannot load the data file	The data file is not in the working directory or has a different name	Make sure the data file resides in the working directory and the name is not misspelled
		The Excel file was loaded and saved as a newer format (.xlsx)	Make sure the file suffix in the xlsread MATLAB command is correct
3A i, 3B i	Error message: 'Superscript indices must either be real positive integers or logicals'	Gene names are separated by ',' sign and not ';' sign	Make sure to use the correct sign
4	Error message: 'Dimensions of matrices being concatenated are not consistent'	Gene names are separated by ',' sign and not ';' sign.	Make sure to use the correct sign
3A i, 3B i, 8, 9, 11, 12, 15	Statement produces an error	Quotation marks are not formatted correctly	Make sure to use opening and closing single quotation marks