EI2022 ISS-199

Capture Optimization for Composite Images

Henry Dietz & Dillon Abney

University of Kentucky

Aggregate.Org UNBRIDLED COMPUTING

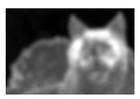
Abstract

A composite image is an image created by combining portions of multiple separately-captured images. Stitching of captures of tiled portions of a larger scene can be used to produce a single composite image (a panorama) with a wider view angle and higher total resolution. Image stacking is a different type of compositing, in which the scene is not changing significantly across captures, but camera parameters might be systematically varied. Focus stacking can extend the depth of field, aperture stacking can implement apodization shaping the out-of-focus point spread function, and noise and motion reduction can be accomplished even using the same camera parameters for each capture to be stacked. These and other compositing methods are well known and commonly used, but the same fixed pattern is commonly used for ordering of captures and choice of capture parameters. This paper examines the problem of static, pseudo-static, or dynamic determination of the optimal capture parameters and ordering.

Key Ideas

- Types of composite images:
  - **Panoramas**, in which entire camera moves
  - **Scans**, in which sensor moves in image plane
  - **Stacks**, usually for ***HDR*** or ***noise reduction***, but also ***super-resolution***, ***moving subject blur reduction***, ***apodization***, etc.
  - **Bracketing**, to obtain “best” of several captures

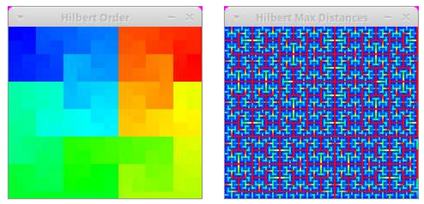
- **For spatially-tiled composites** (panoramas and scans):
  - Use ***walk order*** **tuned to motion hardware abilities**
  - Dynamic scenes are more complex than static ones
    - **Minimize probability of content inconsistencies**
    - **Can use overview/summary to classify regions**

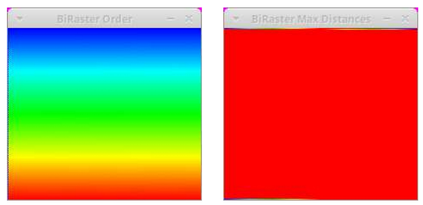
    - **Detect inconsistency and schedule resampling**
  - Scheduler for walk order just needs to be fast enough to keep-up with physical motion & sampling rate...

- **For temporal composites** (stacks and bracketing):
  - Schedule exposure sequence based on analysis of scene and desired result properties
  - Issue is usually settings and number of shots, not ordering within the sequence
  - *Not the main topic of the current work…*

Raster avg. distance 168, median 224

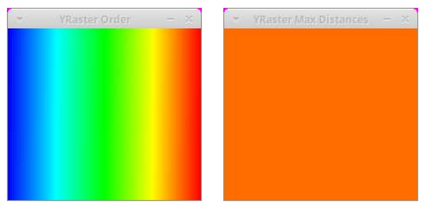


Hilbert avg. distance 201, median 13.2



Biraster avg. distance 187, median 114

Divhilbert avg. distance 190, median 13.7



Yraster avg. distance 150, median 200

Key:

Order is colored from blue (first) to red (last)

Max. is colored from blue (1) to red (maximum)

Probability that a sample is inconsistent varies in proportion to maximum temporal difference from a spatial neighbor and median temporal distance from all spatial neighbors

Static Walk Orders

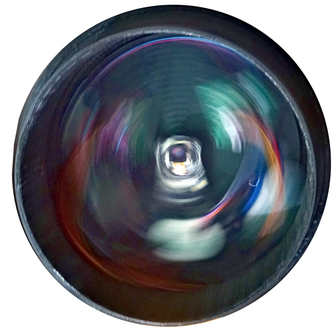
- **Static scene:** minimize ***scan time*** or ***accumulated positioning error***
- **Unknown scene:** minimize ***temporal gap between spatial neighbors***

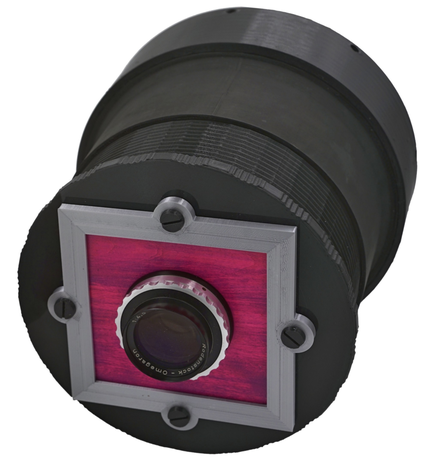
Pseudo-Static Walk Orders

- Start by capturing one or more (fast, low quality) full-scene images
  - Can use ***secondary wider-angle camera*** or ***sparse sampling***
  - Areas with less detail (e.g., sky, defocus) are inherently less critical
  - Areas where nothing changed (e.g., wall vs. tree leaves) are static
- Use a **Genetic Algorithm** (**GA**) to pre-compute static scan ordering for for this scene, ***minimizing temporal skew only where it matters***

Dynamic Walk Orders

- Start scanning using a static or pseudo-static order of a ***to-do list***
- In real time, as each sample is acquired, approximately stitch as per ***“Senscape: modeling and presentation of uncertainty in fused sensor data live image streams”*** from EI2020, `https://doi.org/10.2352/ISSN.2470-1173.2020.14.COIMG-392`
- Incremental stitch gives **confidence** based on overlap between samples; **low confidence** ⇒ scene changed, need to rescan this spot!
  - Add re-sampling of affected pixels to the to-do list
  - Update the scan order, e.g., using a **GA** to reorder to-do list

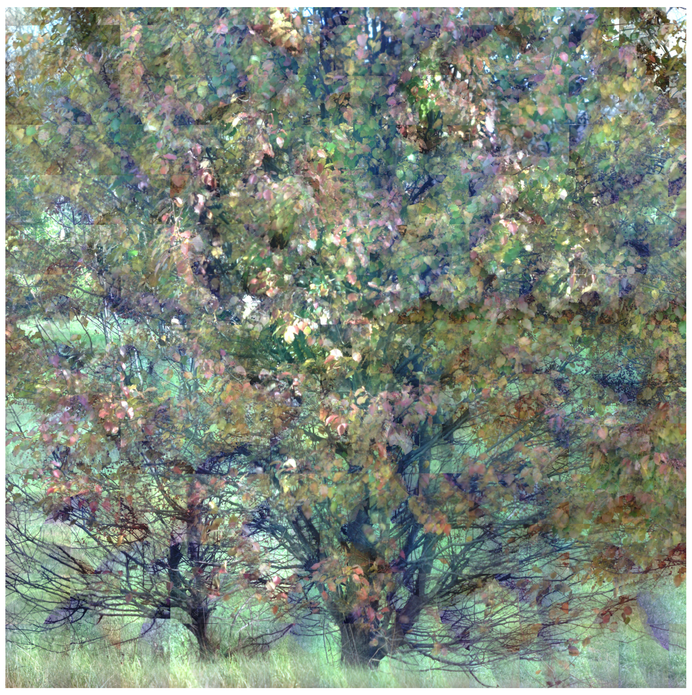




Lafodis:

LArge FOrmat DIgital Scanner

- **Scan resolution:** default **500MP @ 4x5 inch**; theoretical peak **2.6GP**
- **Dynamic range:** 8-10EV; theoretical HDR limit is 20EV
- **Color:** RGB CFA, removable NIR filter
- **Construction:** 3D-printed body, herringbone gear drive, electronics mounts, lens extension, lens focus thread, lens mount plate
- **Electronics:** **ESP32-CAM** and **Arduino Pro Micro** controlling two **28BYJ-48 steppers** with **ULN2003** drivers
- **Scan ordering:** wireless by host, **dynamic angle/radius walk**



Approximate Real-Time Stitch

- Hilbert-based scan ordering; C++, OpenCV control & stitch
- Sample angle and scale are accurate; just searches X,Y position
- **Confidence-based stitch: low confidence** ⇒ it moved, rescan!