Progressive Growing of GANs for Improved Quality, Stability, and Variation

We describe a new training methodology for generative adversarial networks. The key idea is to grow both the generator and discriminator progressively: starting from a low resolution, we add new layers that model increasingly fine details as training progresses. This both speeds the training up and greatly stabilizes it, allowing us to produce images of unprecedented quality, e.g., CelebA images at 1024^2. We also propose a simple way to increase the variation in generated images, and achieve a record inception score of 8.80 in unsupervised CIFAR10. Additionally, we describe several implementation details that are important for discouraging unhealthy competition between the generator and discriminator. Finally, we suggest a new metric for evaluating GAN results, both in terms of image quality and variation. As an additional contribution, we construct a higher-quality version of the CelebA dataset.
Today

- Structure-from-Motion
- Global Optimization
- Bundle Adjustment
- Multi-view Stereo
- Dense Reconstruction
Stereo, we know:
- Extract
- Match
- E or F
- R, t
- 3D points.

How do we add another view?

What information do we already have?
Add Another Camera | The DLT

2D 3D relationship:

\[
\lambda \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_1 & r_2 & r_3 & t_x \\ r_4 & r_5 & r_6 & t_y \\ r_7 & r_8 & r_9 & t_z \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}
\]

Assume we know K, reduce to:

\[
\lambda \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \begin{bmatrix} r_1 & r_2 & r_3 & t_x \\ r_4 & r_5 & r_6 & t_y \\ r_7 & r_8 & r_9 & t_z \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}
\]

Normalized coordinates

We can find lambda...

\[
\lambda = r_7X + r_8Y + r_9Z + t_z
\]
Structure From Motion

Many more things to think about:

- How to pick the best baseline? (to start from)
- How to make sure we don’t accumulate big error with every new camera we add?
- How to correspond points in more than 3 views?
- How to reconstruct geometry from point clouds?

Next class... (Today :)

- **Bundle adjustment - global optimization**
- Simultaneous Tracking & Mapping
Global Optimization | Feature Detection

Detect features using SIFT [Lowe, IJCV 2004]
Match features between each pair of images
Global Optimization | Feature Matching

Refine matching using RANSAC to estimate fundamental matrix between each pair

[Snavely]
Global Optimization | Feature Matching
Problem: Not all points are visible in all images.

We must work incrementally, and stitch the solution together - global optimization.

[Lazebni]
Global Optimization

- Given: $m$ images of $n$ fixed 3D points
  \[
  \lambda_{ij} x_{ij} = P_i X_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n
  \]
- Problem: estimate $m$ projection matrices $P_i$ and $n$ 3D points $X_j$ from the $mn$ correspondences $x_{ij}$
Global Optimization

Start from 2 views:
- Perform reconstruction:
  - Estimate emotion
  - Triangulate 3D points

Add another view:

What's the first step?
Global Optimization

Start from 2 views:

- Perform reconstruction:
  - Estimate emotion
  - Triangulate 3D points

Add another view:

- Find $[R \mid t]$ using 2D-3D correspondence

Now what?
Global Optimization

Start from 2 views:
- Perform reconstruction:
  - Estimate emotion
  - Triangulate 3D points

Add another view:
- Find $[R | t]$ using 2D-3D correspondence
- Triangulate additional points.
Global Optimization

Start from 2 views:

- Perform reconstruction:
  - Estimate emotion
  - Triangulate 3D points

Add another view:

- Find $[R | t]$ using 2D-3D correspondence
- Triangulate additional points.
- Refine the entire reconstruction - **Bundle Adjustment**
Incremental SfM

Figure 7.11 Incremental structure from motion (Snavely, Seitz, and Szeliski 2006) © 2006
Bundle Adjustment

Reprojection error:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} W_{ij} \left\| \frac{1}{\lambda_{ij}} P_i X_j - x_{ij} \right\|^2$$

visibility flag: is point j visible in view i?
Bundle Adjustment

\( g \) is hard to minimize:

- Nonlinear: rotation, perspective division, etc.
- Many (many, many) parameters:
  - 3 for each 3D point, (3+3) for each camera, ...
  - May be order of \( 10^6 \) parameters
- Difficult to initialize

Can add Intrinsic calibration as well (\( K \) matrix for each or all cameras) into optimization.

Solution can be obtained by nonlinear least-squares, e.g. Levenberg-Marquardt.

\[
g(X, R, T) = \sum_{i=1}^{m} \sum_{j=1}^{n} w_{ij} \cdot \left\| P(x_i, R_j, t_j) - \begin{bmatrix} u_{i,j} \\ v_{i,j} \end{bmatrix} \right\|^2
\]

Minimize sum of squared reprojection errors:

predicted image location
observed image location

[Snavely]
Bundle Adjustment
Questions
Multiple View Stereo

Input: calibrated images from several viewpoints
Output: 3D object model
Multiple View Stereo

Snavely
Multiple View Stereo

Source: Y. Furukawa
Multiple View Stereo

Source: Y. Furukawa
Multiple View Stereo

Source: Y. Furukawa

[Lazebni]
Multiple View Stereo | Best Baseline

- Pick a reference image, and slide the corresponding window along the corresponding epipolar lines of all other images, using inverse depth relative to the first image as the search parameter.

Multiple View Stereo | Best Baseline

- For larger baselines, must search larger area in second image
Multiple View Stereo | Best Baseline

Triangulation problem:

What’s the optimal baseline?
- Too small: large depth error
- Too large: difficult search problem
Disparity

Easier to see like this:

\[ d_2 > d_1 \]

Insight?
Why Inverse Depth and not disparity? (like we saw)

SSD with disparity:

The SSD value \( e_{d(i)} \) over a window \( W \) at a pixel position \( x \) of image \( f_0(x) \) for the candidate disparity \( d(i) \) is defined as

\[
e_{d(i)}(x, d(i)) = \sum_{j \in W} (f_0(x + j) - f_i(x + d(i) + j))^2
\]

The real disparity:

\[
d_{r(i)} = \frac{B_i F}{z}
\]

Expected value: (some error assumed)

\[
E[e_{d(i)}(x, d_{r(i)})] = \sum_{j \in W} (f(x + j) - f(x + d(i) - d_{r(i)} + j))^2 + 2N_w\sigma^2_n
\]

But if ...

Suppose that the intensity signal \( f(x) \) has the same pattern around pixel positions \( x \) and \( x + a \)

\[
f(x + j) = f(x + a + j), \quad j \in W
\]

where \( a \neq 0 \) is a constant. Then, from (6)

\[
E[e_{d(i)}(x, d_{r(i)})] = E[e_{d(i)}(x, d_{r(i)} + a)] = 2N_w\sigma^2_n.
\]

It’s ambiguous - produces false match: \( d_{r(i)} + a \)

Remember ambiguity in the DSI...
Now, let us introduce the inverse distance $\zeta$ such that
$$\zeta = \frac{1}{z}.$$  
Take ambiguous case again ($x + a$):
$$E[e_{\zeta(i)}(x, \zeta_r)] = E[e_{\zeta(i)}(x, \zeta_r + \frac{a}{B_i F})] = 2N_w \sigma_n^2.$$  
Still a false match:  
$$\zeta_f = \zeta_r + \frac{a}{B_i F}.$$  
But, dependent on $B_i$ - the baseline.

So we can compound multiple baselines, and find the correct matching point.

When we hit the right match point

$$E[e_{\zeta(i)}]$$
Multiple View Stereo | Best Baseline

Fig. 5. SSD values versus inverse distance: (a) $D = 1h$, (b) $D = 2h$, (c) $D = 3h$, (d) $D = 4h$, (e) $D = 5h$, (f) $D = 6h$, (g) $D = 7h$, (h) $D = 8h$. The horizontal axis is normalized such that $50F = 1$.

Fig. 6. Combining two stereo pairs with different baselines.

Fig. 7. Combining multiple baseline stereo pairs.
Multiple View Stereo | Best Baseline

I1 → I2 → I10

[Okutomi]
[Lazebni]
Questions
MVS at scale

[Snavely et al 2007]
Final Projects

Grade component: 30%

You may work in groups of up to 3 people.

Expected results:

- Documentation: a 2-page description
- Demonstration:
  - Working code
  - Online, e.g. Jupyter notebook
  - Video, screen recording
  - Results with statistical reasoning
  - Kaggle (or equivalent) entry

There will not be a public poster presentation.

Project Option:

- You may take the final exam instead of doing a project
- You may take the final as well as do a project, and the higher grade between the two will be picked
Wrap Up

What we learned today:

- SfM: Global Optimization
- SfM: Bundle adjustment
- MVS: Best Baseline