CoInGP: Convolutional Inpainting with GP

Domagoj Jakobovic, Luca Manzoni, Luca Mariot, Stjepan Picek, Mauro Castelli

l.mariot@tudelft.nl

GECCO 2021, 10–14 July 2021
Motivation: Image Inpainting

- **Problem**: given a damaged image with missing pixels, how can we fill them to obtain a plausible reconstruction?

Existing approaches:

- **Exemplar-based methods** [EL99]
- **Diffusion-based techniques** [BSCB00]
- **Deep learning** (CNNs [LRSWTC18], GANs [IZZE17])
Motivation: Image Inpainting

**Goal**: Investigate GP as a *convolutional inpainting* technique

**Research Questions**:

1. Can GP learn the distribution of pixels in *complete images*?
2. Can GP reconstruct a single degraded image by training on the available pixels?
**Main idea:** sliding window over the image, the surrounding pixels are used to predict the central one with a GP tree

Moore neighborhood:

\[
N_{i,j} = \begin{pmatrix}
x(i-1,j-1) & x(i-1,j) & x(i-1,j+1) \\
x(i,j-1) & ? & x(i,j+1) \\
x(i+1,j-1) & x(i+1,j) & x(i+1,j+1)
\end{pmatrix}
\]
**Main idea:** sliding window over the image, the surrounding pixels are used to predict the central one with a GP tree

**Von Neumann neighborhood:**

\[ N_{i,j} = \begin{pmatrix} x_{(i-1,j-1)} & x_{(i-1,j)} & x_{(i-1,j+1)} \\ x_{(i,j-1)} & ? & x_{(i,j+1)} \\ x_{(i+1,j-1)} & x_{(i+1,j)} & x_{(i+1,j+1)} \end{pmatrix} \]
Supervised Learning Tasks

- **Task 1**: Learn the pixels’ distribution in a dataset of complete images.
- **Training and Test sets**: random samples of MNIST.
- Each pixel retained as correct label.

- **Task 2**: Restore pixels in a single damaged image.
- Random pixel removal with complete neighborhoods.
- **Training set**: available pixels.
- **Test set**: removed pixels.
Fitness Function

- **Training set 1**: $T_1 = \bigcup_{k=1}^{n} F_k$, where $F_k$ is the set of fitness cases of an $M \times N$ image $I_k$ in the MNIST sample:

  $$F_k = \{(N_{i,j}, x_{i,j}) : 1 < i < M, 1 < j < N\}$$

- **Training set 2**: $S \rightarrow$ missing pixels, $P \rightarrow$ available pixels:

  $$T_2 = \{(N_{i,j}, x_{i,j}) : (i,j) \in P, 1 < i < M, 1 < j < N\}$$

- **Fitness Function**: minimize the RMSE between the training set and the GP tree $\tau$ prediction:

  $$fit(\tau) = \sqrt{\frac{\sum_{(N_{i,j}, x_{i,j}) \in T} (\tau(N_{i,j}) - x_{i,j})^2}{|T|}}$$
GP Parameters:

- Functional set: \( \sin, \cos, +, -, /, *, \min, \max, \text{avg}, \sqrt{\cdot} \) and \( \text{pos} \)
- Population size: 500 individuals
- Selection operator: steady-state with 3-tournament operator
- Mutation probability: \( p_m = 0.3 \)
- Termination criterion: 250,000 fitness evaluations
- Linear scaling to clip GP output in the grayscale range \([0, 255]\)

Experimental repetitions in training phase:

- 30 runs for Task 1
- 100 runs for Task 2, for each damaged image
Experiment 1 – MNIST Dataset

▶ **Training set:** random sample of 1 000 images from MNIST

![Histogram](image)

**Moore**

**Von Neumann**

▶ **Main finding:** GP learns the pixels’ distribution significantly better than the baseline (=average of surrounding pixels)
Experiment 2 – Single Images

- Two test images: *Boat* and *Goldhill*, both resized to $256 \times 256$ pixels
- Random removal of 20% of the pixels
- Enforce missing pixels with complete neighborhoods
- Every two columns: first one is kept, 100 non-adjacent pixels are removed from the second
Main finding: GP is able to score lower prediction scores than the baselines, for both neighborhoods.

Moore neighborhood is better than Von Neumann, despite the fact that it can use less fitness cases.
Further finding: the prediction error is concentrated on the edges of the images.
Conclusions & Future Directions

Research Question 1:
- CoInGP can successfully learn the pixels’ distribution in a dataset of complete images, better than baseline predictors

Research Question 2:
- CoInGP can provide plausible inpainted images, with Moore neighborhood working better than Von Neumann

Future work:
- Address the case of *incomplete* neighborhoods
- Consider *multi-layer architectures*
- Compare with state-of the art methods such as CNNs
- Apply convolutional GP to other domains (e.g. text generation [MJMPC20])
Thank you for your attention!


