Faculty of Computer Science

June 17, 2024

Long-range Camera Guiding for the Person Recognition in Public Spaces.

Project Supervisor:
Khelvas Alexander Valerievich

Author:
Timchenko Daniil Gennadievich
dgtimchenko@edu.hse.ru

Introduction

The Project was done in association with Timur Khaibrakhmanov.

- The first objective of the study is to study the generalization of TSP (Travelling Salesman Problem) - Dynamic TSP, where weights of a graph are dynamic. The second objective is to propose and assess the algorithm, that could solve this problem in the context of long-range cameras:

$$\text{Approach unvisited agent} \longmapsto \text{Wait for s iterations} \longmapsto \text{Repeat}$$

- The importance of advanced agent camera tracking systems is undeniable (surveillance, business, automation). Furthermore, close to optimal algorithms for solving the DTSP (Dynamic Travelling Salesman Problem) could help across many various fields (logistics, routing, etc)

TSP is already a NP-Complete problem. Although many methodologies were discovered (dynamic programming, ant colony, Christofides), they provide only with an approximate solution. For finding the optimal one, time complexity can be stated as: $O(\frac{n!}{n})$. When dealing with DTSP, we have to not only consider all options in current time, but also the options across 3-rd dimension - time. This brings even more intricacy and higher time complexity.

Formal Problem Statement

DTSP for People Traversal

Let $\mathbb{R}^3$ be the vector space, and $P_t = \{(x_1^{(t)}, y_1^{(t)}), \ldots, (x_n^{(t)}, y_n^{(t)})\}$ would be a set of observed objects existing in this vector space, that lie on a plane $z = 0$ in moment of time $t \in \mathbb{N}$ $(x, y \in \mathbb{R})$. Let $\mathcal{P}$

$$\mathcal{P}_t(\hat{x}, \hat{y}, \hat{z}, \phi_t, \psi_t, \theta_t,) \to \mathcal{V}$$

be the projection function that calculates corner points of FOV projection $\mathcal{V}_t \in \mathbb{R}^{3\times4}$ on $z = 0$ in the moment t. Here $\hat{x}, \hat{y}, \hat{z}$ - coordinates of a camera, ϕ_t, ψ_t, θ_t - yaw (azimuth), pitch (elevation) and roll in the time moment t (rotation coordinates).
Let $\mathcal{C}$

$$\mathcal{C}(P_t, \mathcal{V}_t) \to \begin{bmatrix} \Delta\phi_t & \Delta\psi_t & \Delta\theta_t \end{bmatrix}^T$$

be the controller function, that makes a decision on the controlling of camera direction and zoom for the time-step $t + 1$. Camera rotations then are updated as following:

$$\begin{bmatrix} \phi_{t+1} \\ \psi_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} \phi_t \\ \psi_t \\ \theta_t \end{bmatrix} + \begin{bmatrix} \Delta\phi_t \\ \Delta\psi_t \\ \Delta\theta_t \end{bmatrix}$$

Formal Problem Statement

Let $\mathcal{I}$

$$\mathcal{I}_t(\mathcal{P}_t, \mathcal{V}_t, p_1, p_2) \to a \in \{0, 1\}$$

Be the indicator function, concluding if an observed object is taking up from p_1 to p_2 portion of space on a viewfinder AND never was observed in proper ratio before ($p_1 \leq p \leq p2$). a in this case is an indicator (True or False). Then the constrained optimization problem looks as following:

$$\begin{cases} \sum_{t=1}^{T} \mathcal{I}_t(\mathcal{P}_t, \mathcal{V}_t, p_1, p_2) \geq n \\ T \to min_{\mathcal{C}} \end{cases}$$

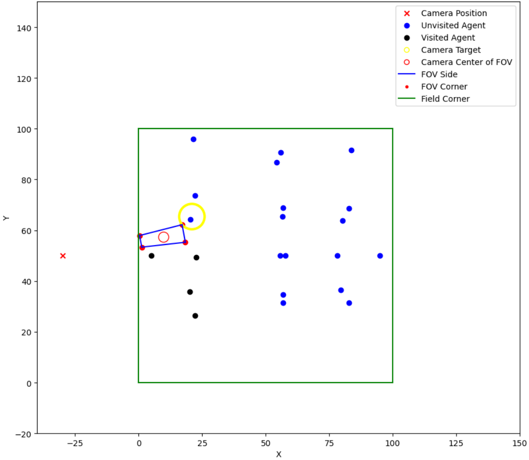
Camera Model

Researched and implemented theory

- Projective Geometry for Coordinate Transformation and Field of View
- Azimuth and Elevation of a camera
- Parametric equations for Line intersections

Algorithmic Development
Methodologies

- Fixed Path (Master-Route)
- Nearest Neighbor Greedy Algorithm



Nearest Neighbor Greedy Algorithm

Intuition behind the algorithm

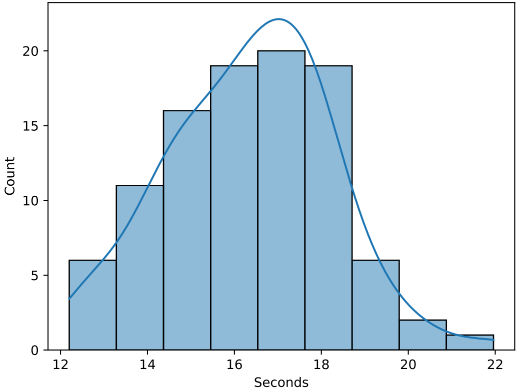
1. find closest unvisited agent to the current camera aim position
2. approach the position of closest unvisited agent
3. wait fot s iterations
4. if agent was able to go outside the FOV, continue approaching him, else go to step (1.)

As a dataset for evaluation, a comprehensive study by Timur Khaibrakhmanov was made in order to create a soccer match simulation, that is representative of real world matches. The long simulation (40 min.) then is used to sequentially evaluate the NN Greedy algorithm. The metric T in this case is the mean time required to traverse all agents at least once.

$$T = \frac{1}{n_{iter}} \sum_{i=1}^{n_{iter}} t_i$$

Empirical Findings
Nearest Neighbor-Greedy

- For $n_{players} = 22$ and 100 experiments with $t_{stop} = 5$
- Average traversal time: 16.34 seconds
- Standard deviation: 1.87 seconds (across 100 simulations)
- A robustness of method is apparent, given the randomized dataset



Contributions:

- Bridging theoretical concepts with practical applications
- Developing a sophisticated computational framework

Future Work:

- Continued research for enhancing algorithm efficiency
- Refinement of accuracy
- Exploring other possible algorithms
- Working on versatility

Discussion