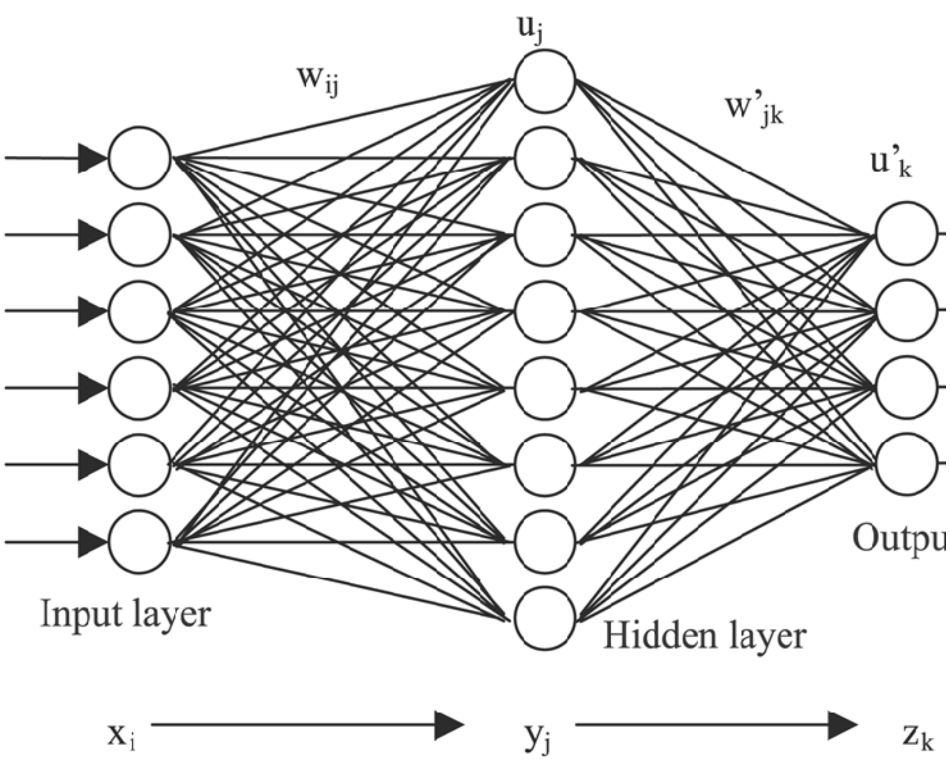
Harnessing the Power of Machine Learning for Improving the Safety of Outer-Space Travel

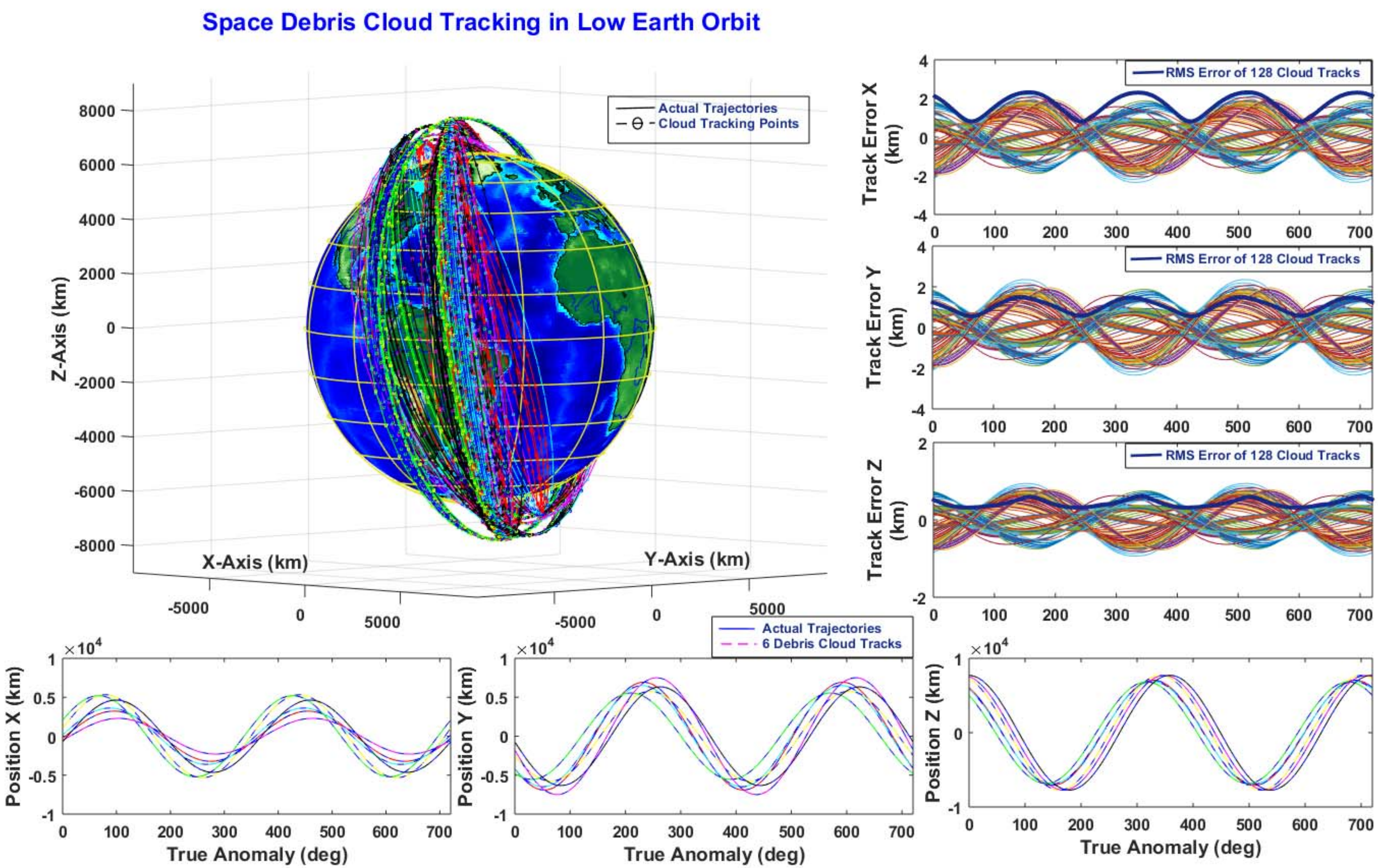
Amber Yang
Stanford, University
MATLAB EXPO 2017

What is Machine Learning?



- Arthur Samuel (1959). Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.
- Artificial Neural Networks (ANNs): computing systems based on connectionism.

Artificial Neural Networks for Classifying and Tracking Space Debris



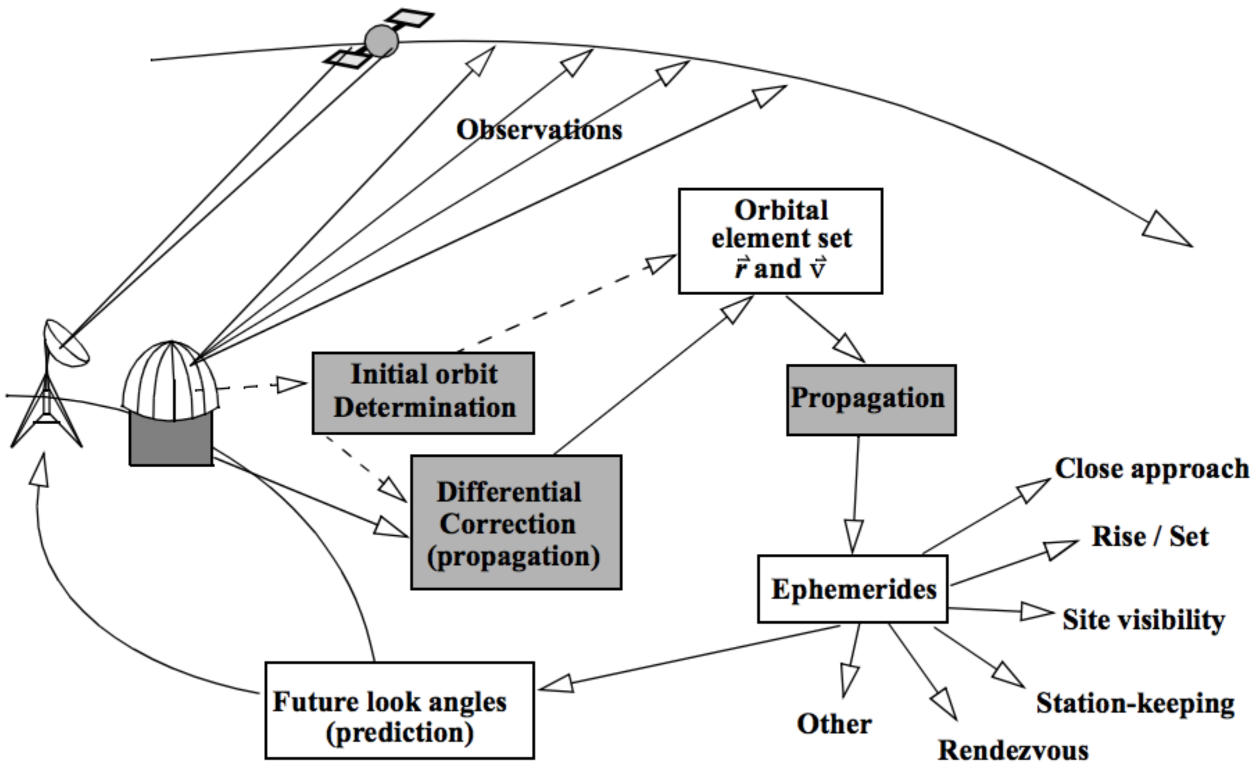
Dangers of Space Debris in Low Earth Orbit

- Millions of space debris in Low Earth Orbit (LEO) pose collision threats to on-orbit spacecraft and satellites[1]
- Kessler syndrome predicts space debris population will increase exponentially—challenging ability to track and catalogue collision threats[2]

Photo Source: European Space Agency

1. B.G. Cour-Palais et. al. 1978
2. D.J. Kessler et. al. 2010

Traditional Orbital Tracking Methods



Demonstration of Extended Kalman Filter; Photo Source: D.A. Vallado et. al. 1998

- Utilization of radar, laser, and optical imagery to identify and observe space debris
- Extended Kalman Filter (EKF): state transition model of error dynamics statistically corrected via error covariance propagation to estimate orbital waypoints for trajectory prediction[3]

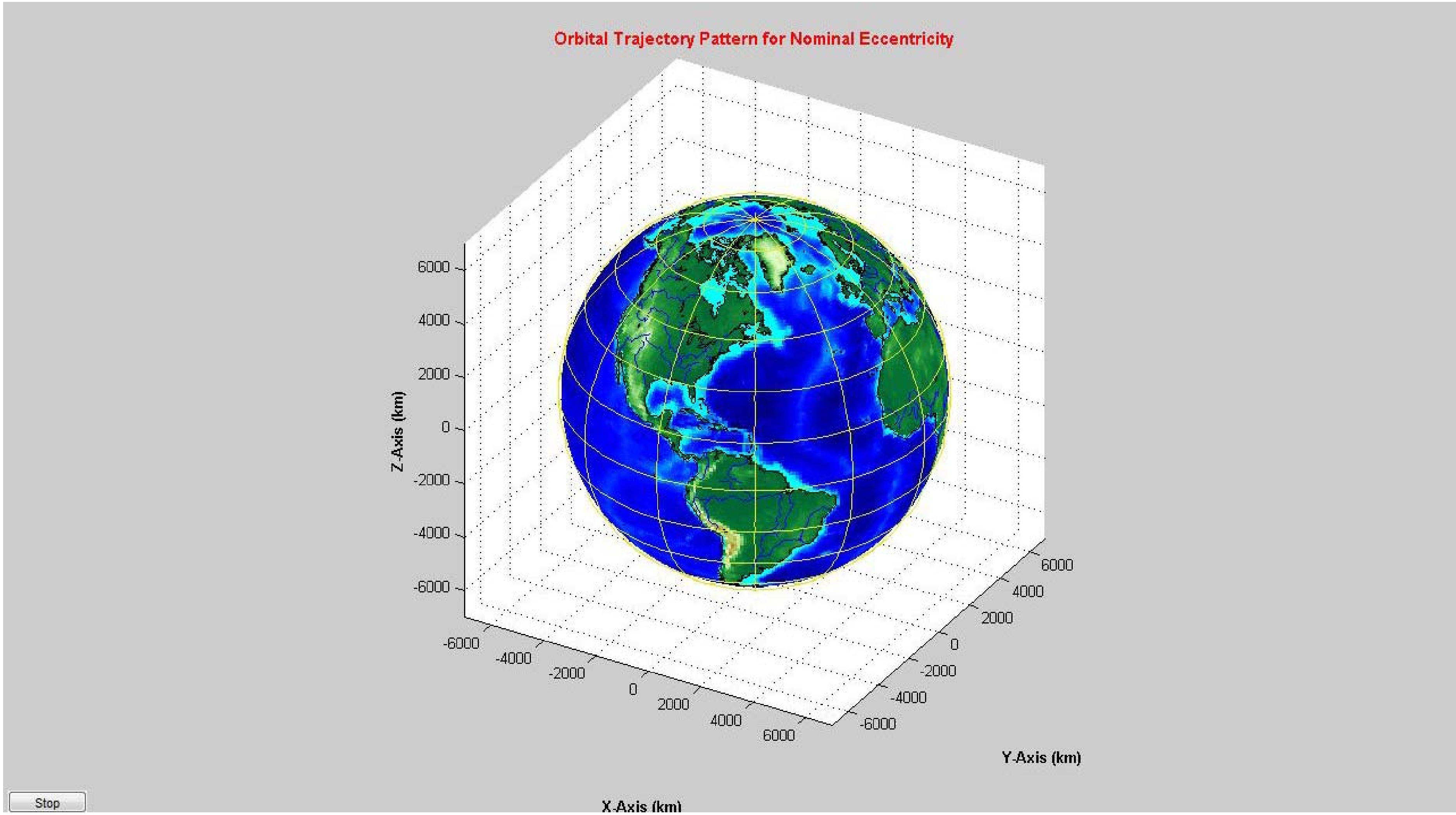
3. R.E. Kalman, 1960

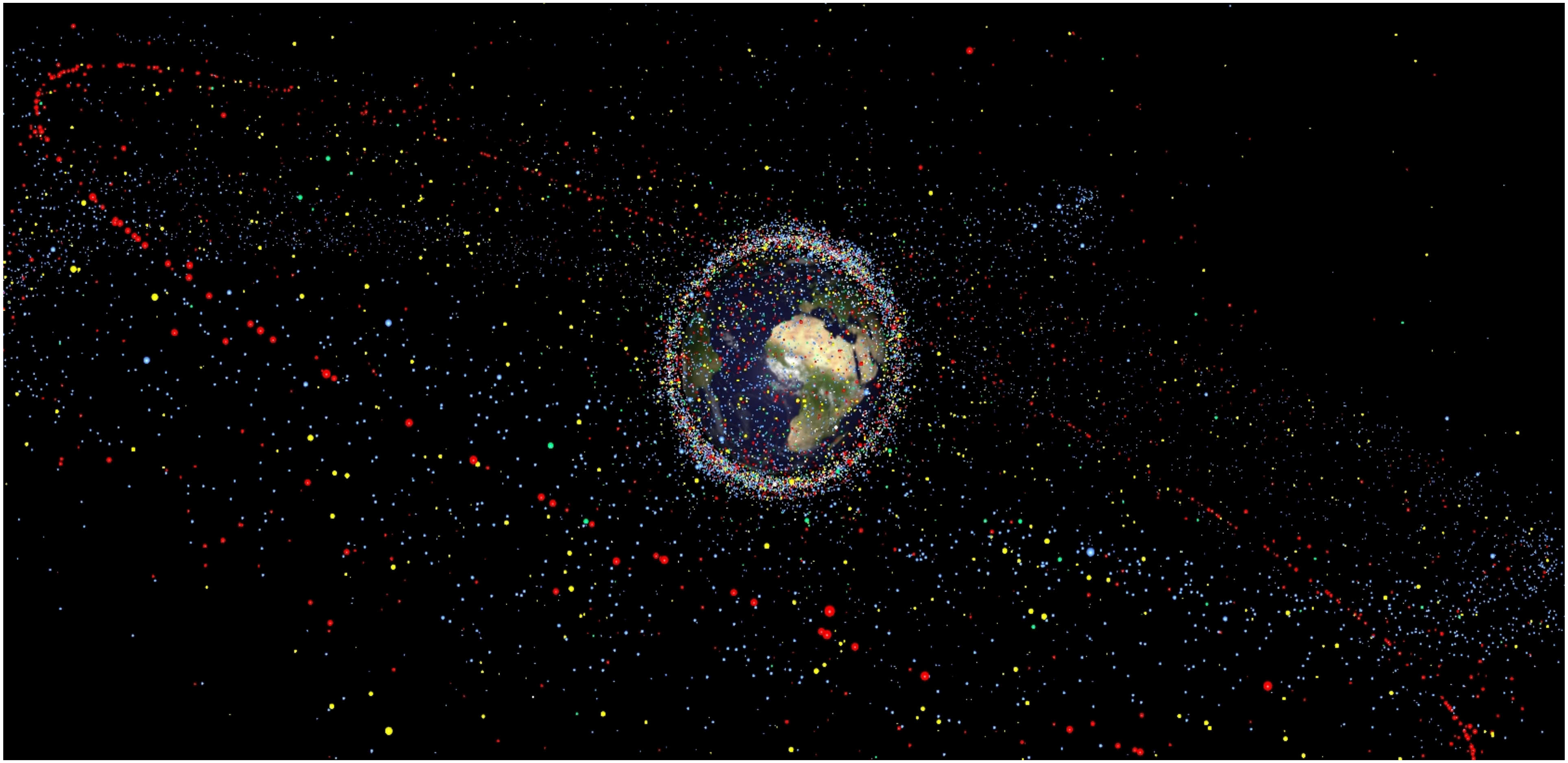
Problem

- Space surveillance network requires detecting, tracking, and cataloguing algorithms all in one comprehensive system
- Space debris is small in size, travels at high speeds, and orbits at high altitudes
  - These characteristics impact EKF tracking accuracy
- Astrodynamics of orbiting objects constantly changing due to celestial disturbances[4]
  - Frequent manual tuning of EKF parameters necessary for off-track space targets
- Without self-learning and training abilities, covariance-driven tracker must be adjusted for individual space targets

4. D.A. Vallado et. al. 1998

Orbital Patterns Recognized within Keplerian Elements

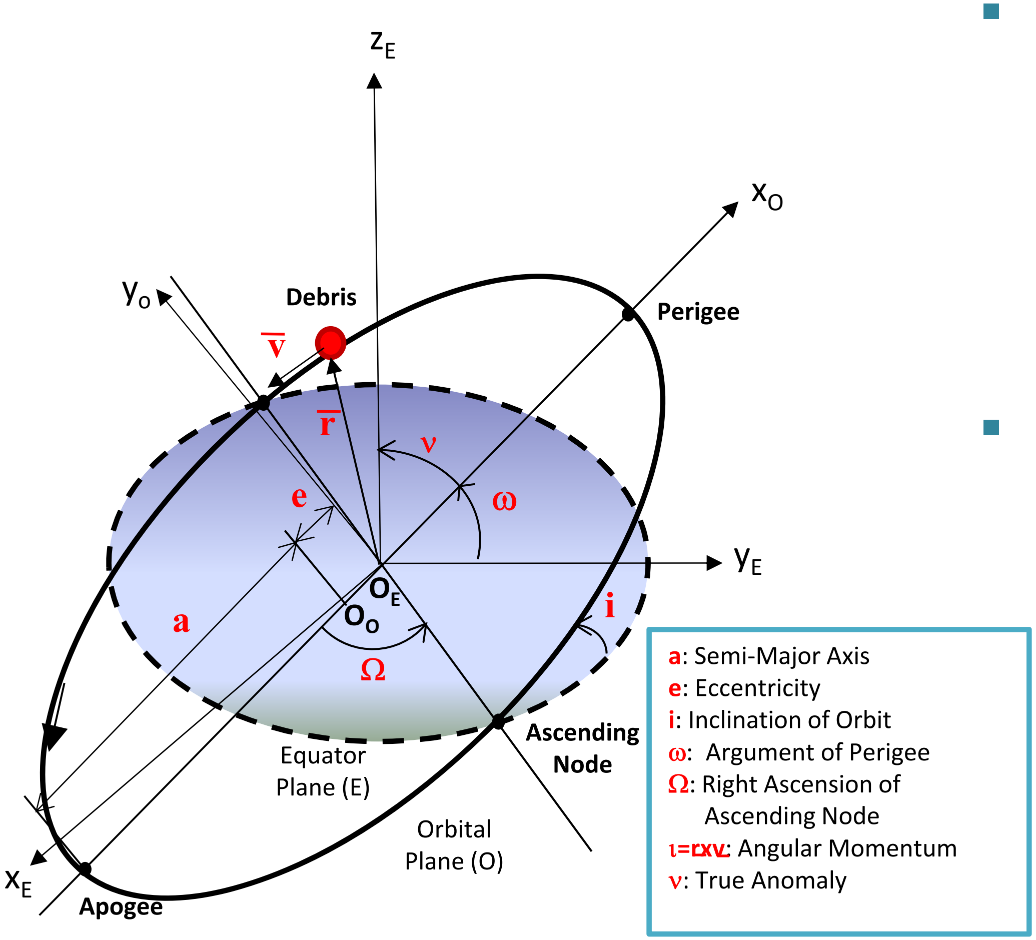




Are there inherent geometrical patterns in the orbits of space debris that can be learned by an Artificial Neural Network for accurate detection and tracking over time?

Phase One of Research: Orbital Recognition for Space Debris Tracking Using Artificial Neural Networks

Theory and Hypothesis



Geometrical Diagram of Keplerian Elements in Orbit

- **Theory**: Study invoked by 2014 Nobel Prize for discovery that the brain can act as an inner-Global Positioning System (GPS) due to its ability to recognize geometric patterns[5]
- **Hypothesis**: If discovery of an inner-brain GPS is applied for an outer-space GPS, pattern recognition Artificial Neural Networks (ANN)[6] can act as a human brain to detect, track, and catalogue orbits of space debris in LEO using Keplerian elements that have inherent geometric patterns

5. O'Keefe et. al. 1978

Kinematic Equations of Keplerian Elements

Each Keplerian element provides a unique geometrical pattern to configure an orbit[11]

Semi-major axis of an orbit (*a*):

$$a = \frac{\mu}{\frac{2\mu}{r} - v^2}$$

where *r* is the magnitude of position and *v* is the magnitude of velocity

Eccentricity vector (*e*):

$$\vec{e} = \frac{1}{\mu}\left[\left(v^2 - \frac{\mu}{r}\right)\vec{r} - (\vec{r} \cdot \vec{v})\vec{v}\right]$$

where $\vec{r}$ and $\vec{v}$ are the position and velocity vectors

Inclination of the orbit (*i*):

$$i = \cos^{-1}\frac{(\vec{\kappa} \cdot \vec{\iota})}{|\vec{\iota}|}$$

where $\vec{\kappa}$ is a unit vector in *z*-component, and $\vec{\iota} = \vec{r} \times \vec{v}$ provides angular momentum, which was selected as an orbital element for pattern recognition in this study

Argument of perigee (ω):

$$\omega = \cos^{-1}\frac{\vec{n} \cdot \vec{e}}{|\vec{n}||\vec{e}|}$$

where $\vec{n}$ is the vector pointing to the ascending node

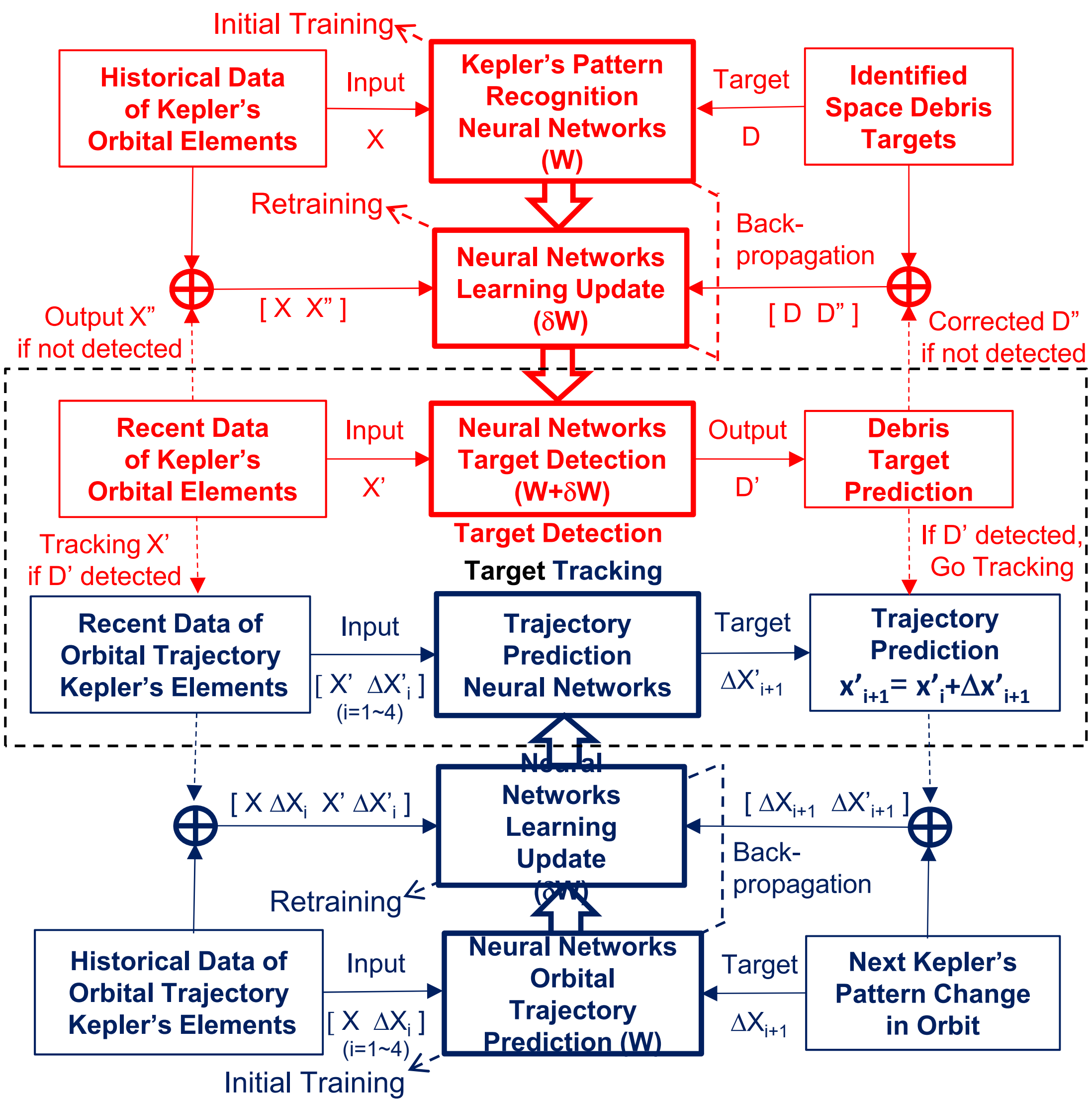
Right ascension of ascending node (Ω)

$$\Omega = \cos^{-1}\frac{n_x}{|\vec{n}|}$$

Since the semi-major axes of objects in LEO are similar to one another, angular momentum is selected to replace the semi-major axis as an orbital element for the neural networks in this research.

11. H.D. Curtis, 2005

Orbital Recognition ANN System Design

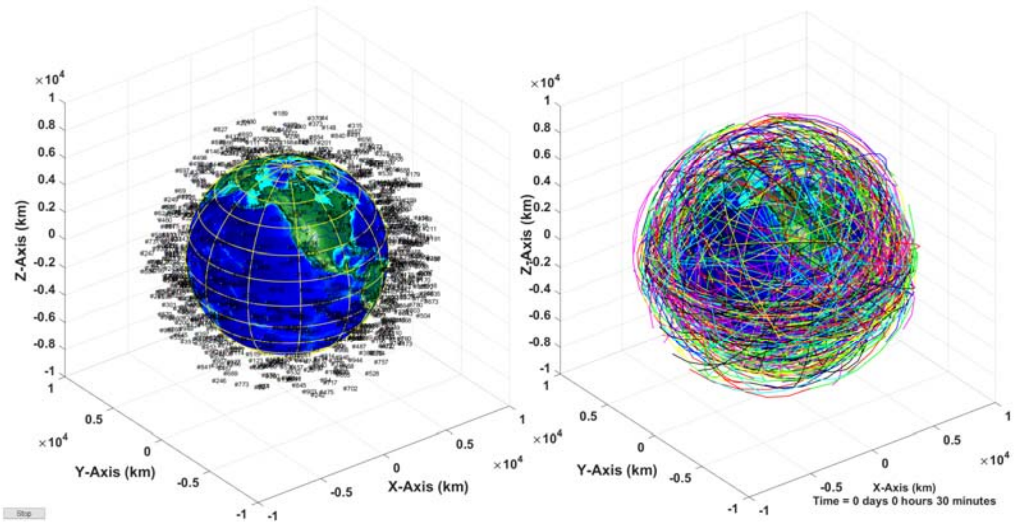


Orbital Recognition System Schematic Diagram

Orbital Recognition System comprised of two ANNs with inputs being five Keplerian elements:

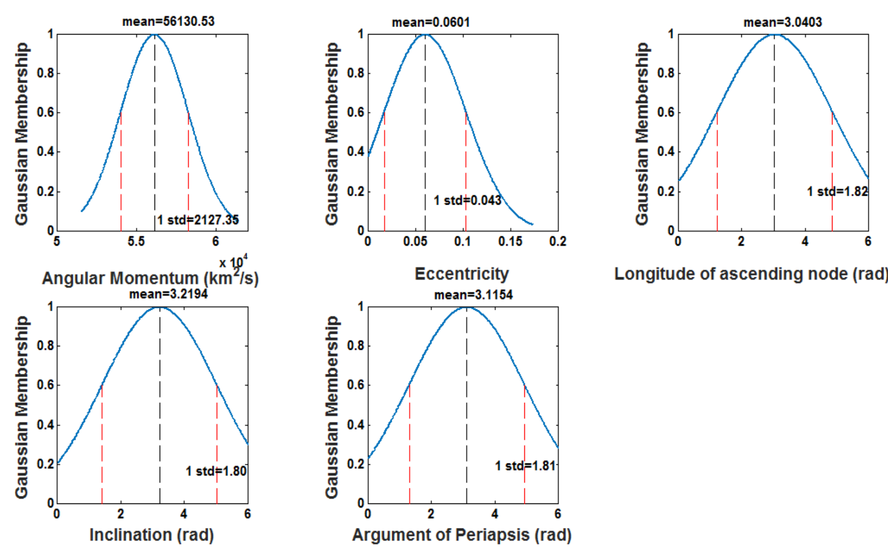
- **Target Detection ANN** (red): subsystem for identification, classification, and cataloguing space debris
- **Trajectory Prediction ANN** (blue): subsystem for monitoring and tracking space debris

Engineering Process
Stage 1: Random Samples of Space Debris via Keplerian Elements

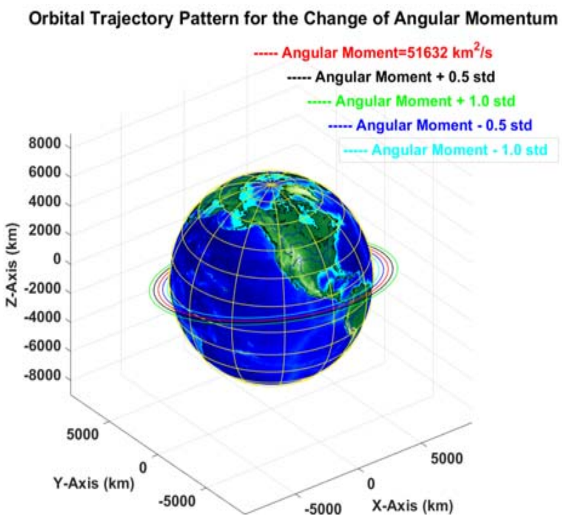


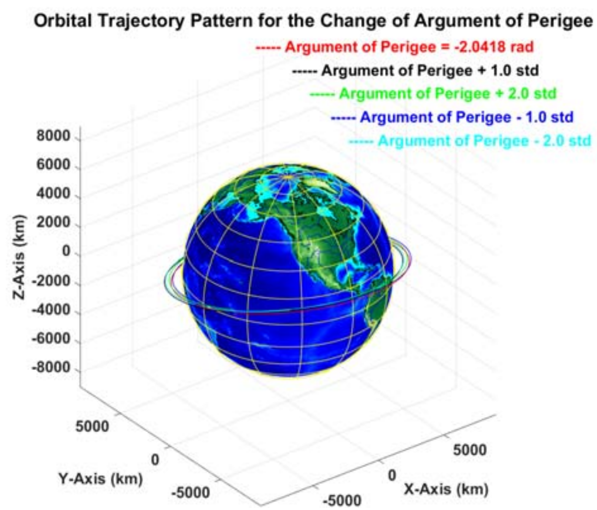
- 1,000 random space debris samples generated in terms of Keplerian elements seen in Earth-fixed coordinates X_E-Y_E-Z_E
- Samples satisfy LEO constraints:
  - Altitude: 200-1,800 km
  - Period 90-120 min
  - Semi-major axis length: Earth's radius + 80 km

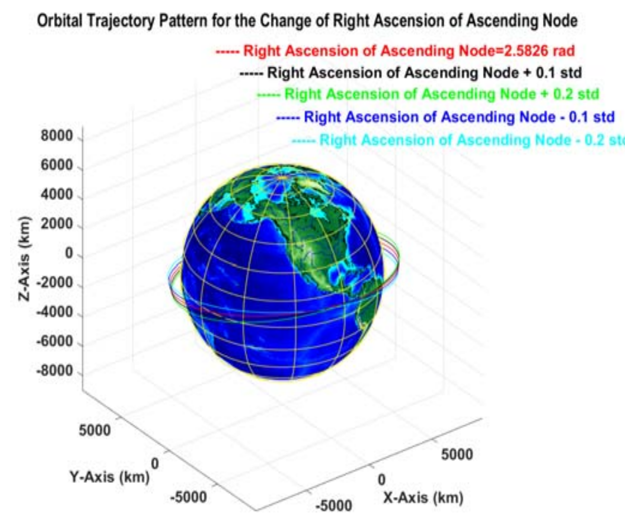
Gaussian membership functions generated for random samples of space debris

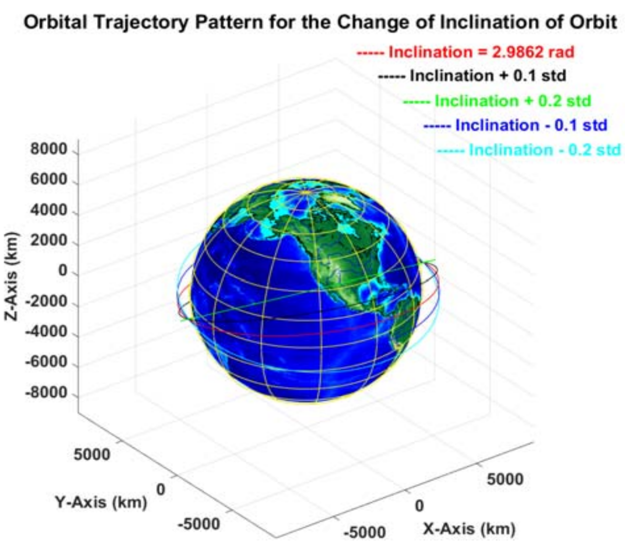


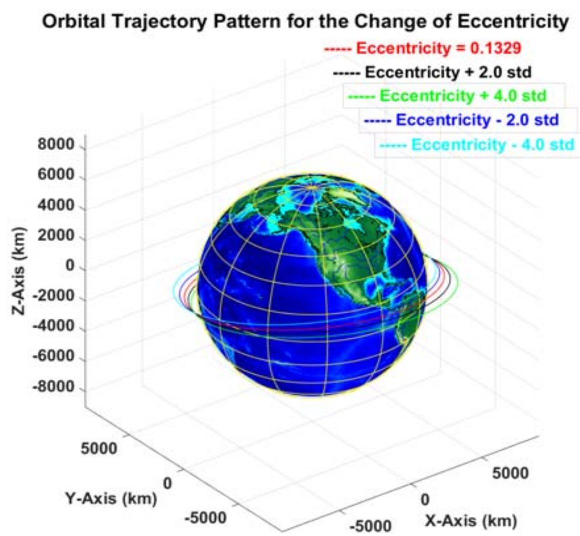
Engineering Process
Stage 2: Orbital Patterns from Keplerian Variations









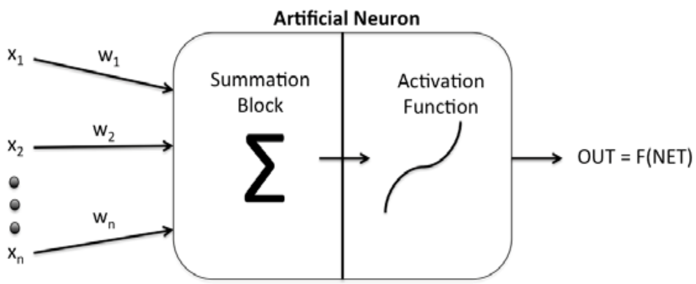


- Keplerian variations imposed on space debris samples to demonstrate changes of orbital patterns in trajectories
- Variations with different STD scales in Keplerian elements provides testing patterns for ANN system

Engineering Process
Stage 3: Implementation of ANN Design

- Backpropagation[8] ANN is commonly used for supervised training

Feedforward:



- Processes data through connecting artificial neurons with links
- Each link has a numeric weight, and weights are updated throughout ANN training process
- Once NET value is calculated, processed by an activation function (hyberbolic-tangent):

$$OUT = \tanh(NET) = \frac{1 - e^{-NET}}{1 + e^{-NET}}$$

Backpropagation:

- As input is propagated through system, each hidden layer of neurons contributes to errors in output layer
- Output error signals are transmitted back from output layer to each neuron in hidden layers
- Process repeated until each neuron in the network has received an error signal that describes its contribution to the overall system error
- Formulae used to update weights, w_{jk}, between neurons *j* and *k* with gradient descent of weights:

$$w'_{jk} = w_{jk} + \Delta w_{jk}$$

$$\Delta w_{jk} = l_r \Delta_k (1 + O_k)(1 - O_k) x_k = l_r \delta_k x_k$$

where $\delta_k = (1 + O_k)(1 - O_k)\Delta_k$ is the modified error

8. P.J. Werbos, 1990

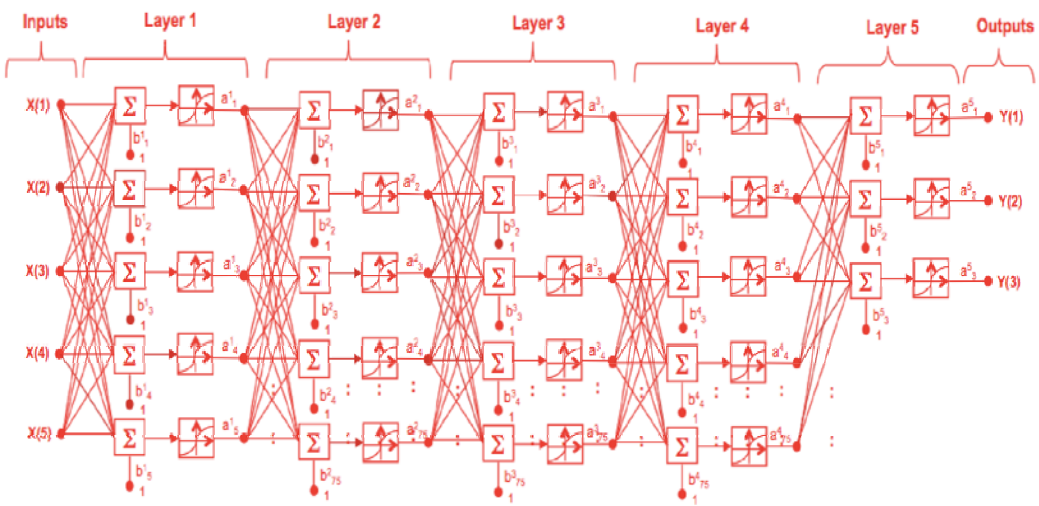
Engineering Process
Stage 4: ANN Orbital Recognition System

	Input Layer	Hidden Layers /Neurons	Output Layer	Training Rate	Weighting Momentum	MSE	Samples
Target Detection ANN	5	3 / 75	3	0.5	0.11	2.00E-3	1000
Retraining Detection ANN	5	3/75	3	0.33	0.33	2.78E-5	Retrained Cases
Trajectory Prediction ANN	25	2 / 10	5	0.4	0.33	1.00E-5	1

Parameters of Target Detection ANN and Trajectory Prediction ANN

- Programming of Orbital Recognition System is coded in MATLAB
- Trade-off analysis of ANN parameters deduces that:
  - At least three hidden layers required for Target Detection of more than 1,000 samples
  - 75 (+/-25) neurons in hidden layers are sufficient to avoid over-and-under-learning for Target Detection
  - Nominal value of training rate and momentum is 0.33 (+/-0.22) for good convergence
  - If MSE is tenfold greater, ANN accuracy will decrease to 80~90%
  - Time of completion in MATLAB on PC is four hours for initial Target Detection ANN, 30 minutes for Retraining Detection ANN for retraining 100 samples, and 10 minutes for Trajectory Prediction ANN

Engineering Process
Stage 5: Target Detection ANN



Target Detection Backpropagation ANN Schematic Diagram

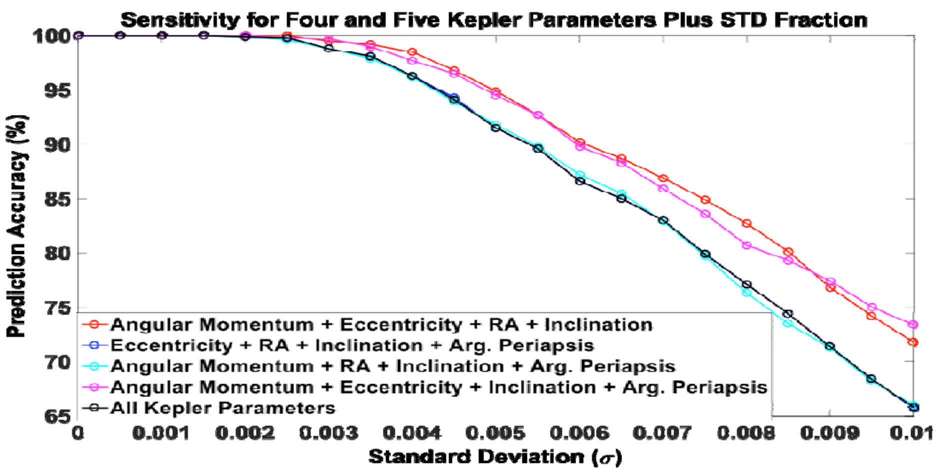
- Mapping system of associating specific orbital patterns and their variations with an identification number (array of identification indices)

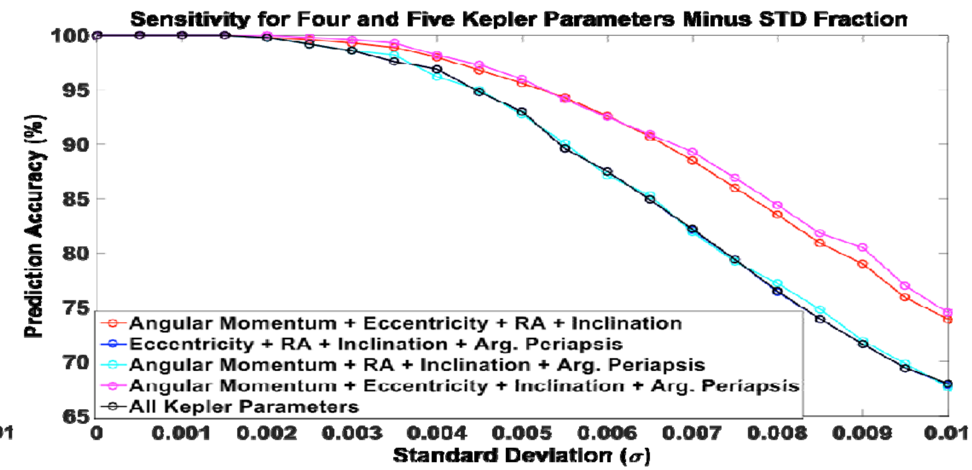
- Input: 1,000x5 matrix of 1,000 space debris in terms of five Keplerian elements
- Output: space debris ID—three indices with values ranging from -1 to 1 at an interval of 0.2
  - 1,000 1x3 ID arrays randomly assigned to 1,000 space debris samples
- **Phase 1 of Target Detection ANN: Initial training**
- **Phase 2 of Target Detection ANN: Testing**

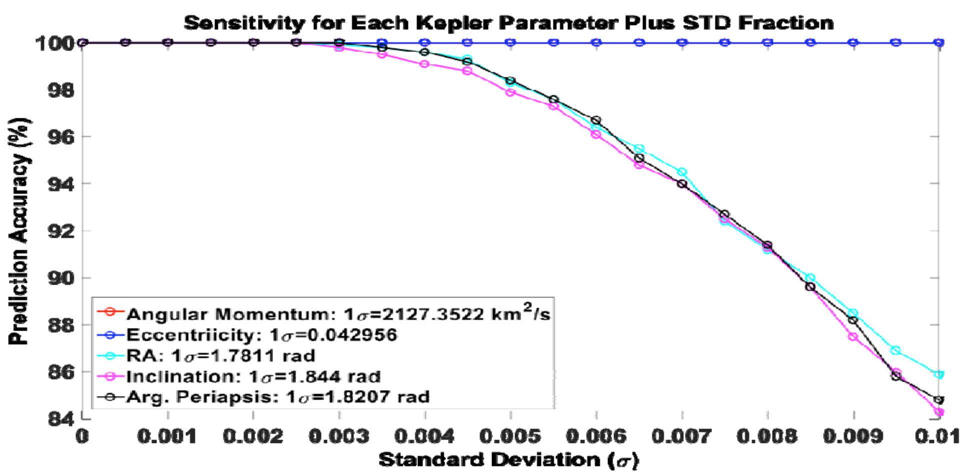
Engineering Process
Stage 6: Retraining Target Detection ANN

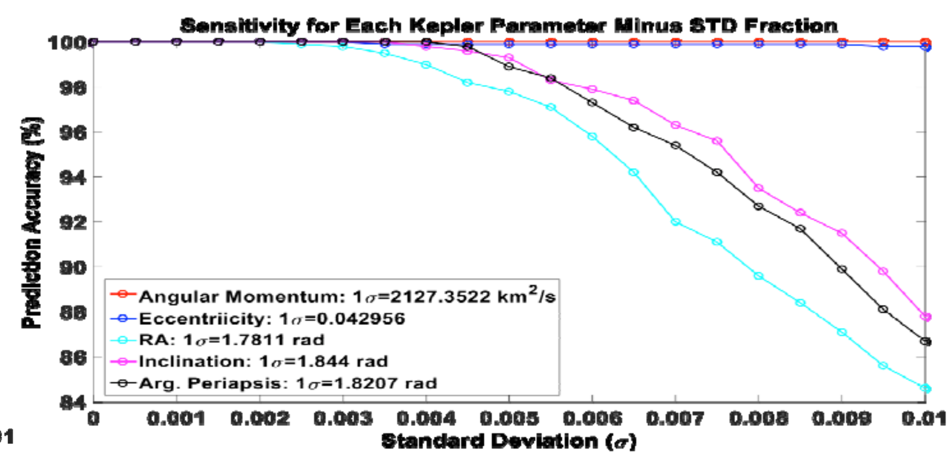
- After correcting for mistargeted samples in Testing Phase, readjustment of ANN weights through backpropagation
- Initial weighting matrices for retraining provided by current ANN system that has been trained
- Weighting matrices are only adjusted for new input patterns with variations that were mistargeted

Results: Target Detection ANN

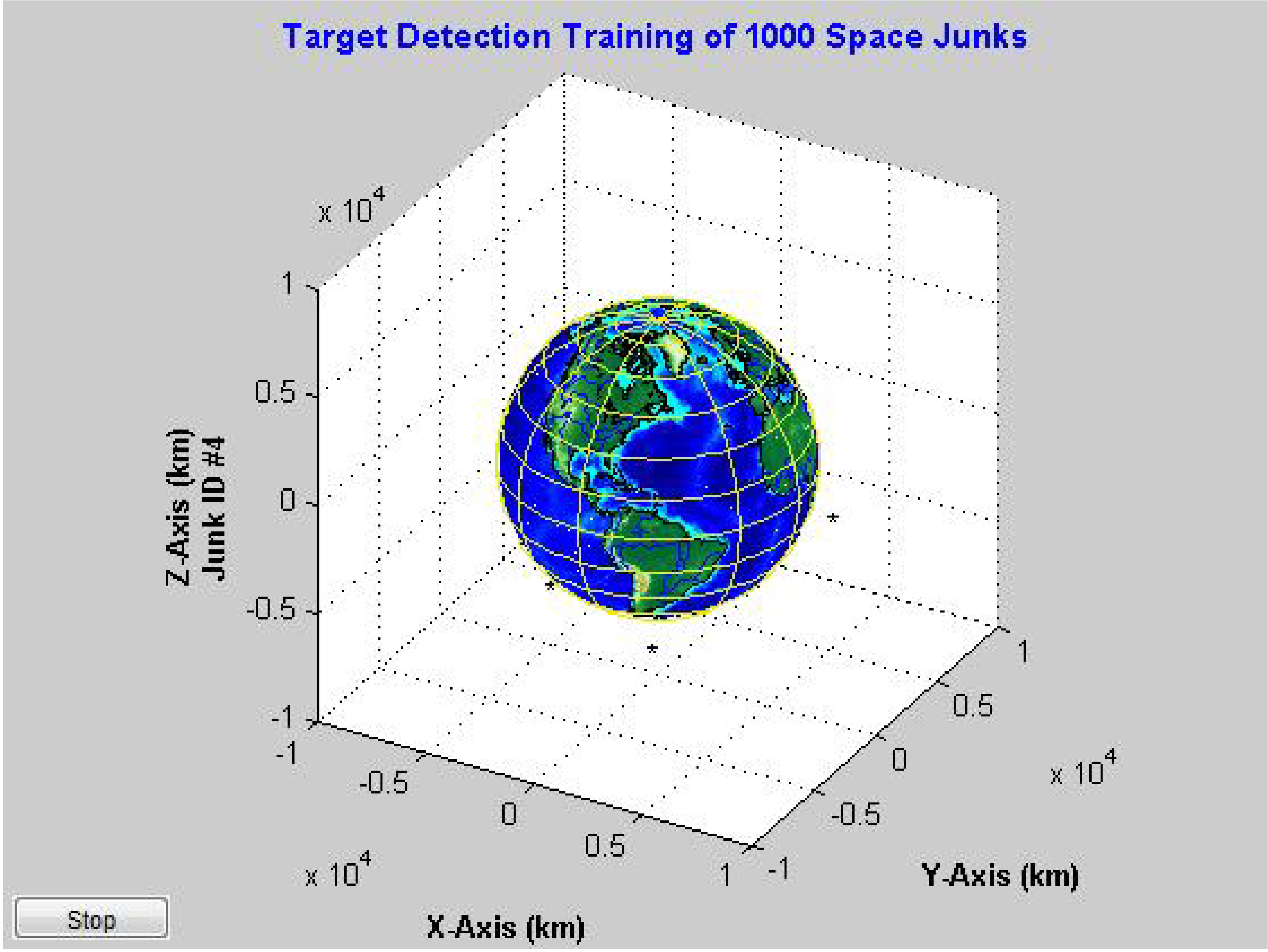




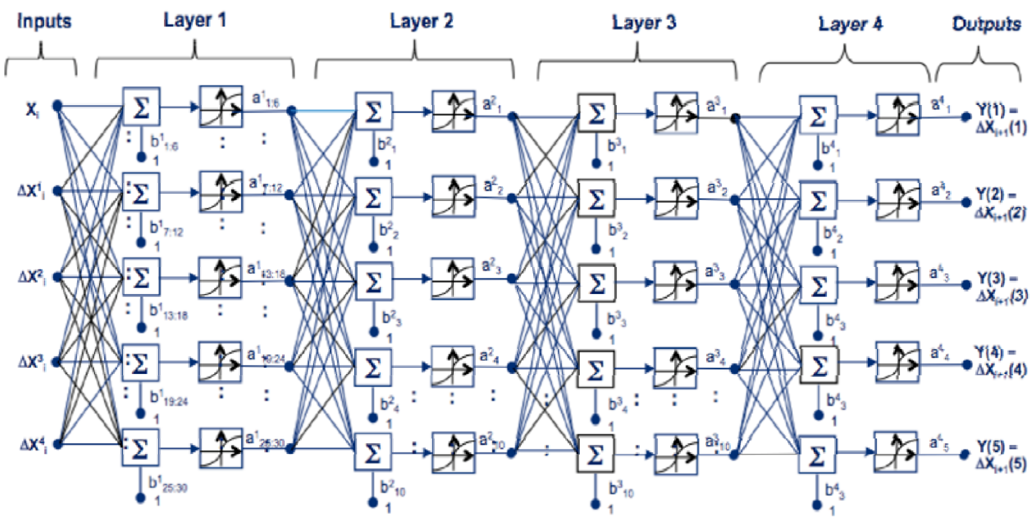




Animation of Target Detection ANN



Engineering Process
Stage 7: Trajectory Prediction ANN



Trajectory Prediction Backpropagation ANN Schematic Diagram

- Changing patterns of Keplerian elements recognized to predict changes of orbital patterns that will likely occur in the next waypoint for orbital prediction

- Initial training samples: Five consecutive waypoints of Keplerian elements used to produce four successive changes of five Keplerian elements to generate twenty changes for last waypoint in sequence (540 samples)

- Testing samples: generated by training input matrix at a different increment of true anomaly, a different starting waypoint, and through waypoints never before trained

Results: Trajectory Prediction ANN

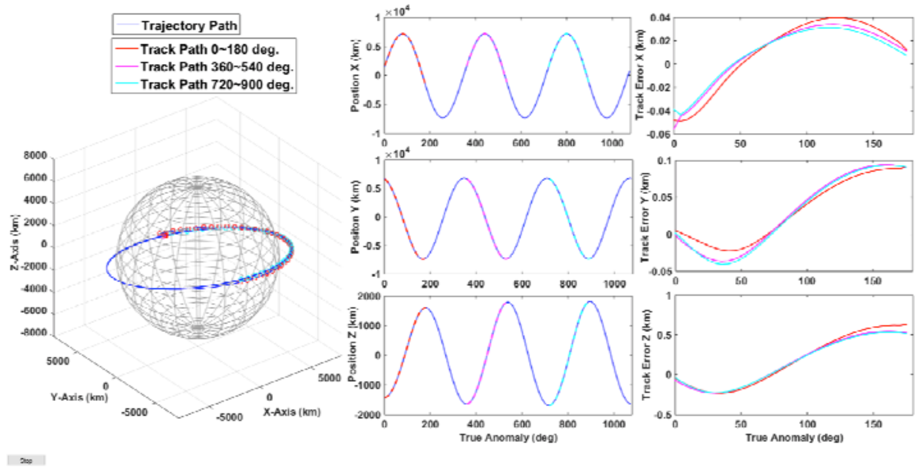


Figure 1: Simulation of training waypoints at every 1-degree true anomaly and tracking from 0-degree true anomaly

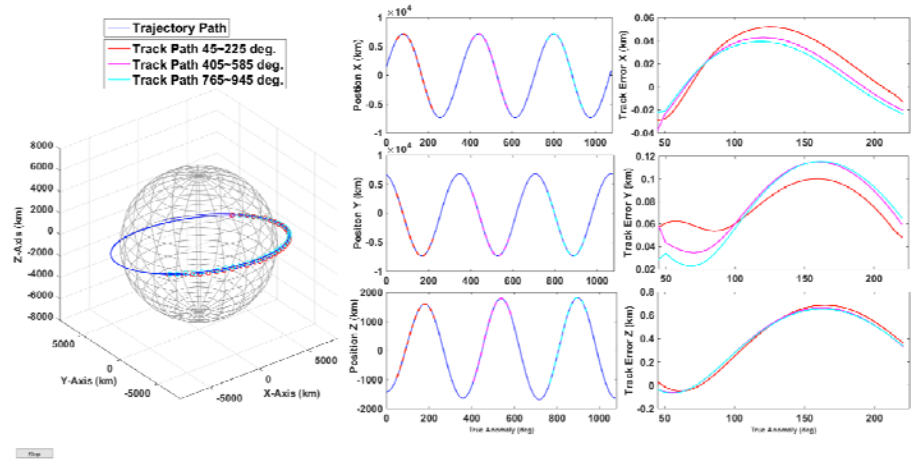


Figure 2: Simulation of training waypoints at every 1-degree true anomaly and tracking shifted 45-degree true anomaly

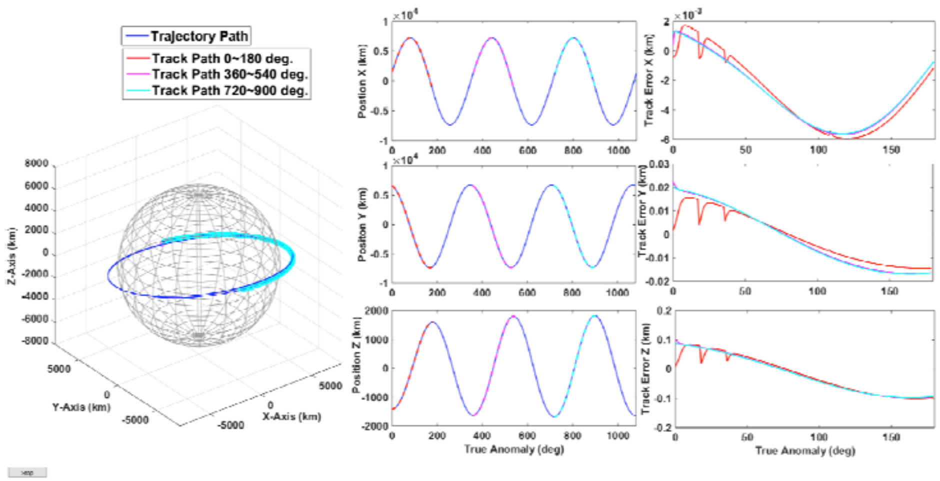


Figure 3: Simulation of training waypoints at every 0.2-degree true anomaly and tracking from 0-degree true anomaly

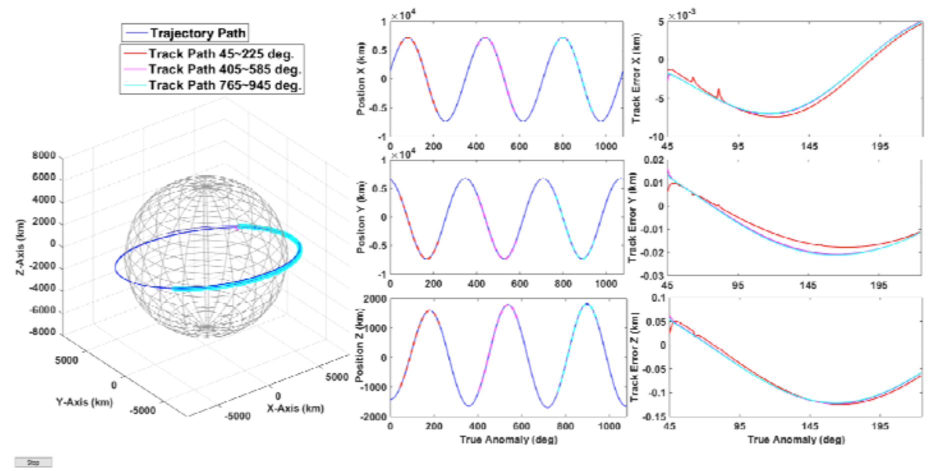
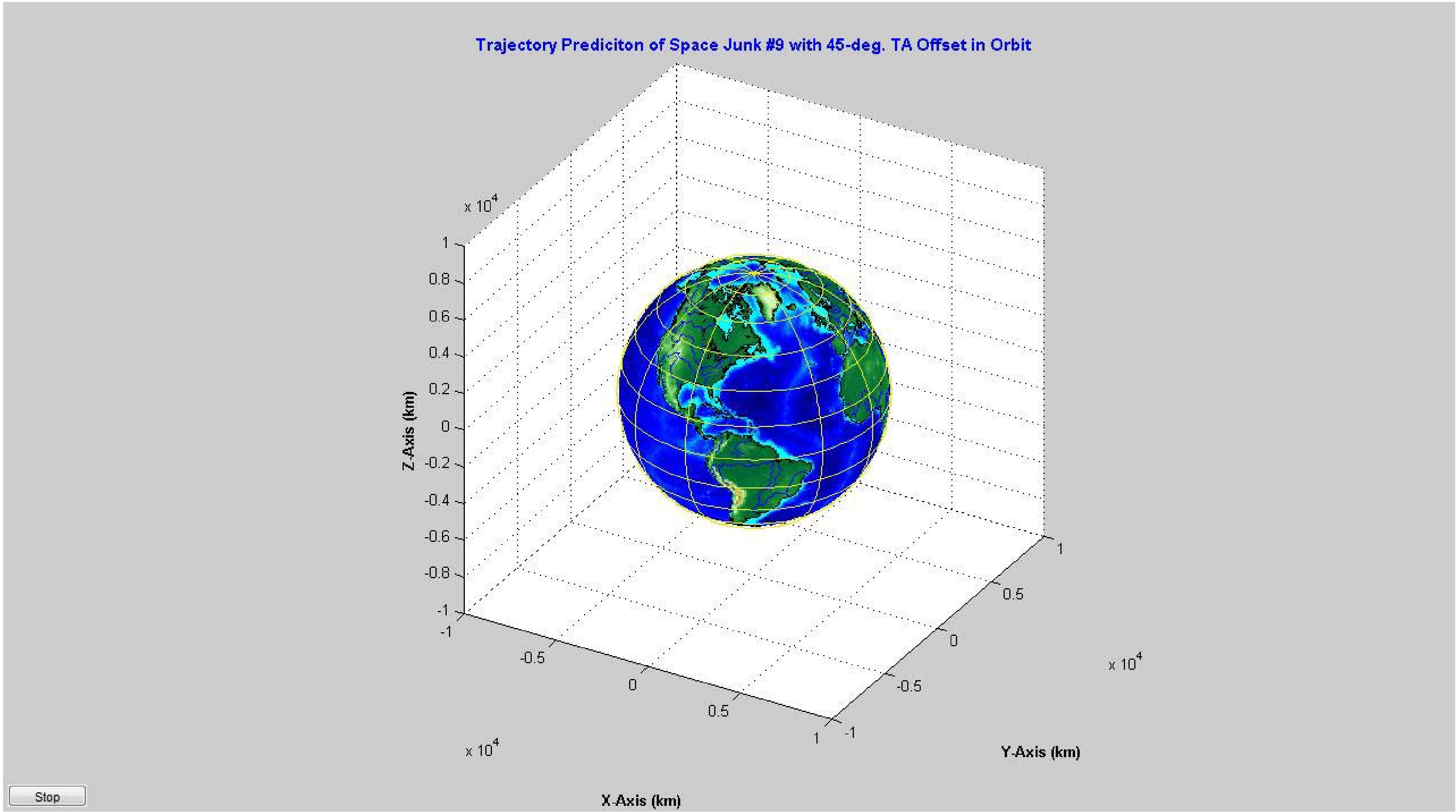
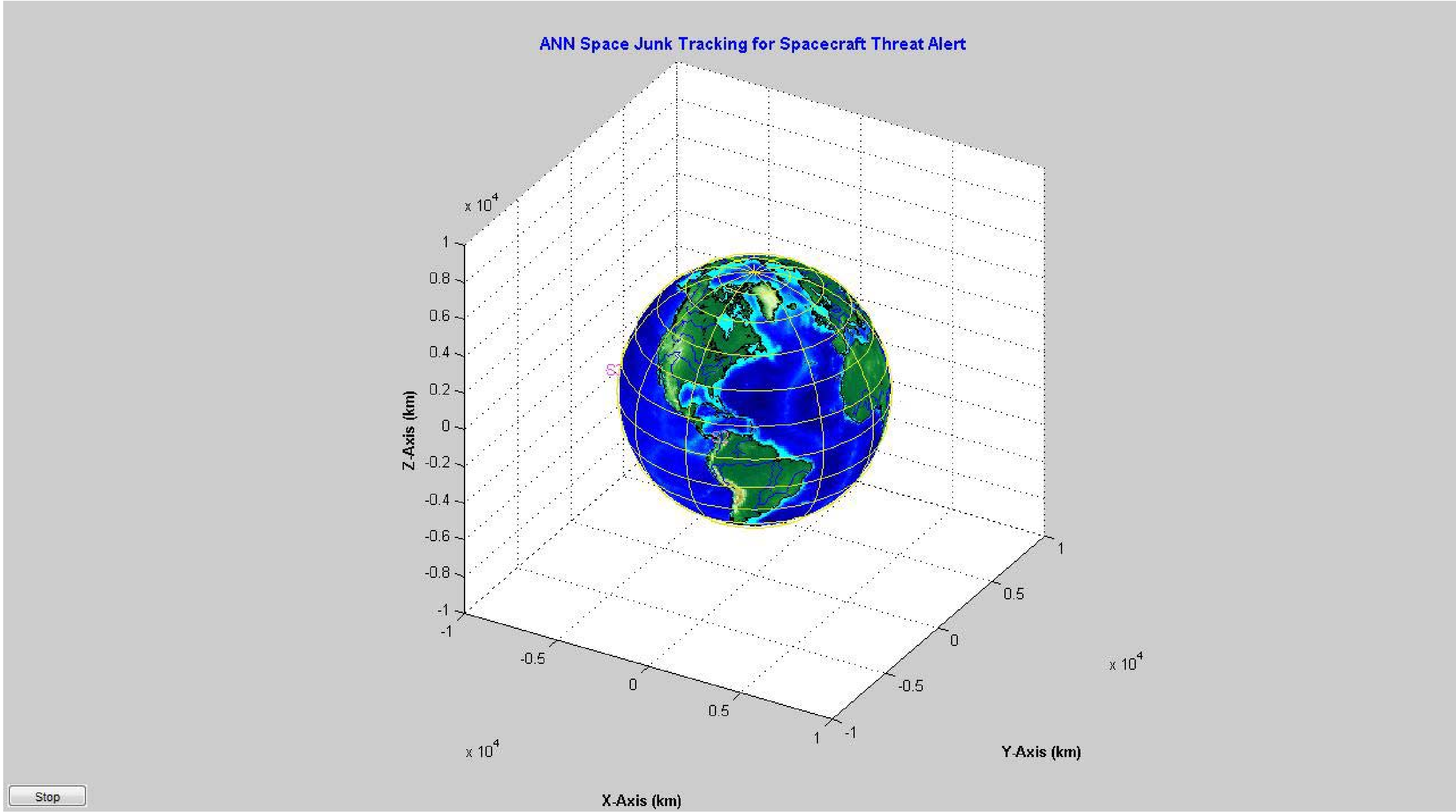


Figure 4: Simulation of training waypoints at every 0.2-degree true anomaly and tracking shifted 45-degree true anomaly

Animation of Trajectory Prediction ANN

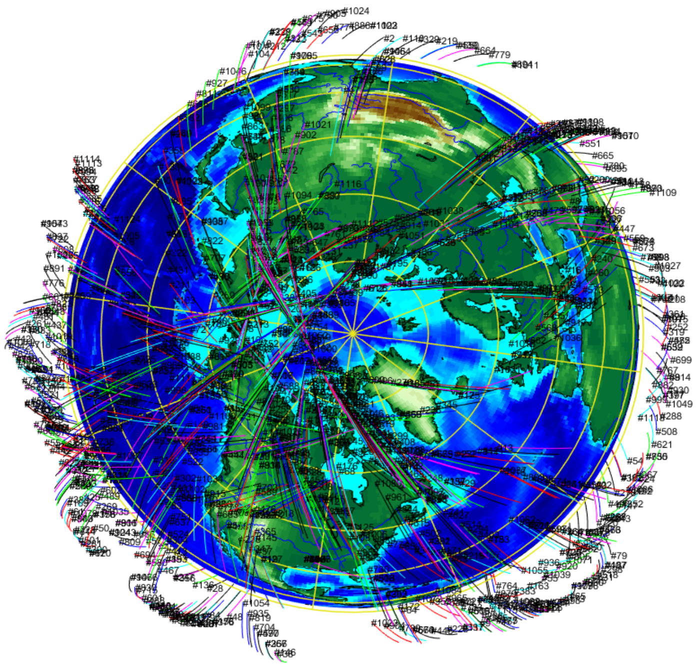


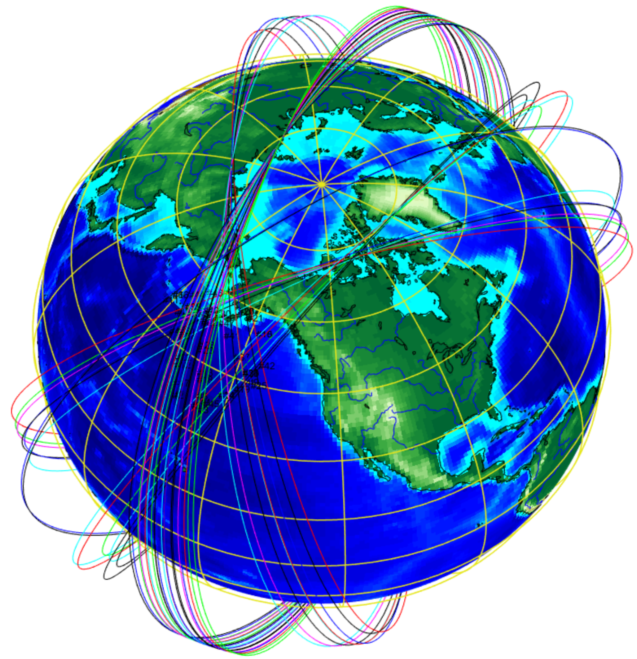
Comprehensive Space Debris Collision Avoidance System



Phase Two of Research: Multi-Orbit Space Debris Cloud Tracking Using Iterative Closest Points Registration with Machine Learning

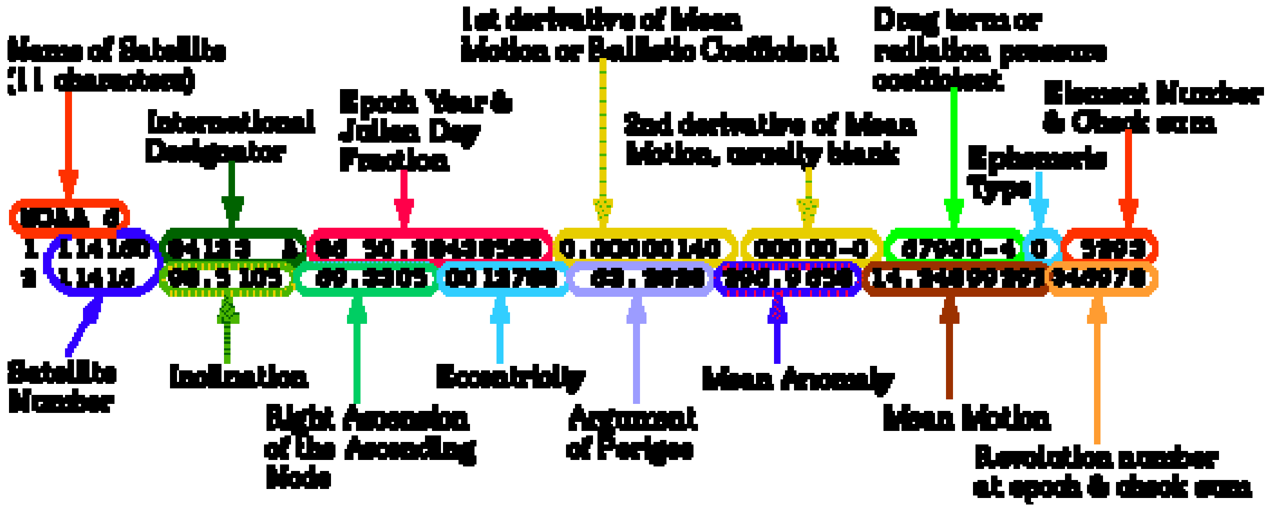
Observed Space Debris Cloud Development





- Orbital data that was collected from May 27-June 5, 2016 for 2559 space debris are analyzed to identify the clouds of space debris orbiting in close vicinity to each other.
- A space debris sample was chosen as the center of the selected debris cloud, and other space debris samples within a distance of 3500 km and an inclination of 0.5 to 1.5 radians around the cloud center were included to form a space debris cloud.

Implementation of Real Space Debris Data for Space Debris Cloud Tracking



State Transition Equation

$$\begin{bmatrix} \delta \boldsymbol{r}(t) \\ \delta \boldsymbol{v}(t) \end{bmatrix} = \Phi(t, t_o) \begin{bmatrix} \delta \boldsymbol{r}(t_0) \\ \delta \boldsymbol{v}(t_o) \end{bmatrix}$$

Keplerian State Transition Matrix

$$\Phi = \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{21} & \Phi_{22} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \boldsymbol{r}}{\partial r_o} & \dfrac{\partial \boldsymbol{r}}{\partial r_o} \\ \dfrac{\partial \boldsymbol{v}}{\partial v_o} & \dfrac{\partial \boldsymbol{v}}{\partial v_o} \end{bmatrix}$$

where $\Phi_{11}, \Phi_{12}, \Phi_{21}, \Phi_{22}$ *are functions of the Keplerian elements* $\boldsymbol{a}, \boldsymbol{e}, \boldsymbol{i}, \boldsymbol{\omega}$, *and* $\boldsymbol{\Omega}$.

Definition: Iterative Closest Points Registration

- Iterative Closest Point (ICP) algorithm: method to register 3D data for geometric alignment between two independent scans in one frame of reference.
- ICP converges to the nearest local minimum of mean-square distance at a fast rate of convergence within a few iterations.

$$E(\boldsymbol{R},\boldsymbol{T}) = \sum_{i=1}^{n}\sum_{j=1}^{m}\omega_{ij}\left[\!\left[m_i - (\boldsymbol{R}d_j + \boldsymbol{T})\right]\!\right]^2$$

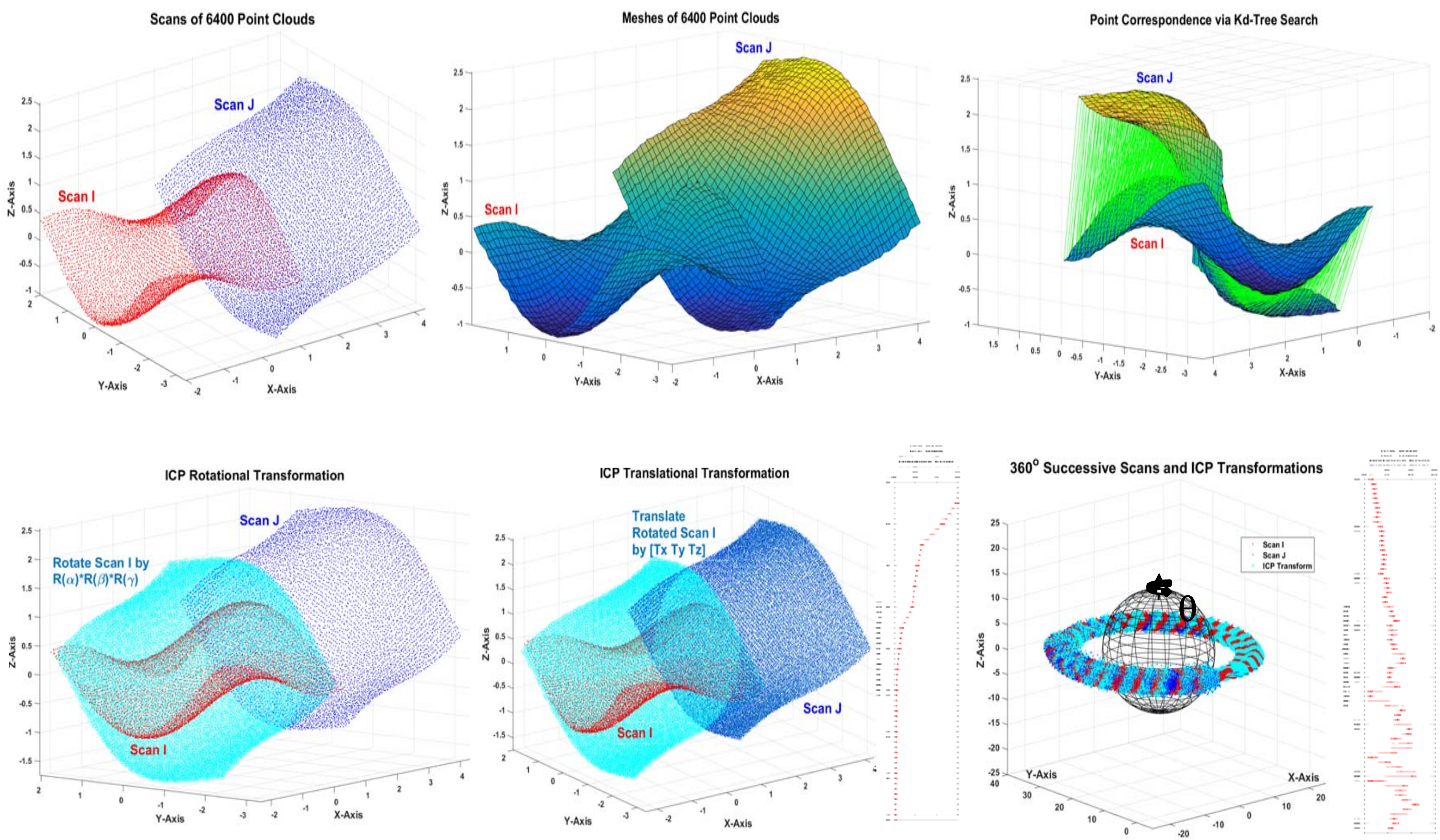
- The Singular-Value-Decomposition (SVD) algorithm is applied for the ICP Alignment function to determine a rotational matrix $\mathbf{R}_{\mathbf{abg}}$ and a translation vector $\mathbf{T}_{\mathbf{xyz}}$ for a rigid transformation of two meshed point clouds form *i*-scan to *j*-scan to reach for a minimum of the point-to-plane mean-square distance metric.

$$\boldsymbol{R} = \boldsymbol{R_x}(\alpha)\boldsymbol{R_y}(\beta)\boldsymbol{R_z}(\gamma) = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos\alpha & -\sin\alpha\\ 0 & \sin\alpha & \cos\alpha\end{bmatrix}\begin{bmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{bmatrix}\begin{bmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{bmatrix}, \boldsymbol{T} = \begin{bmatrix}T_x\\ T_y\\ T_z\end{bmatrix}$$

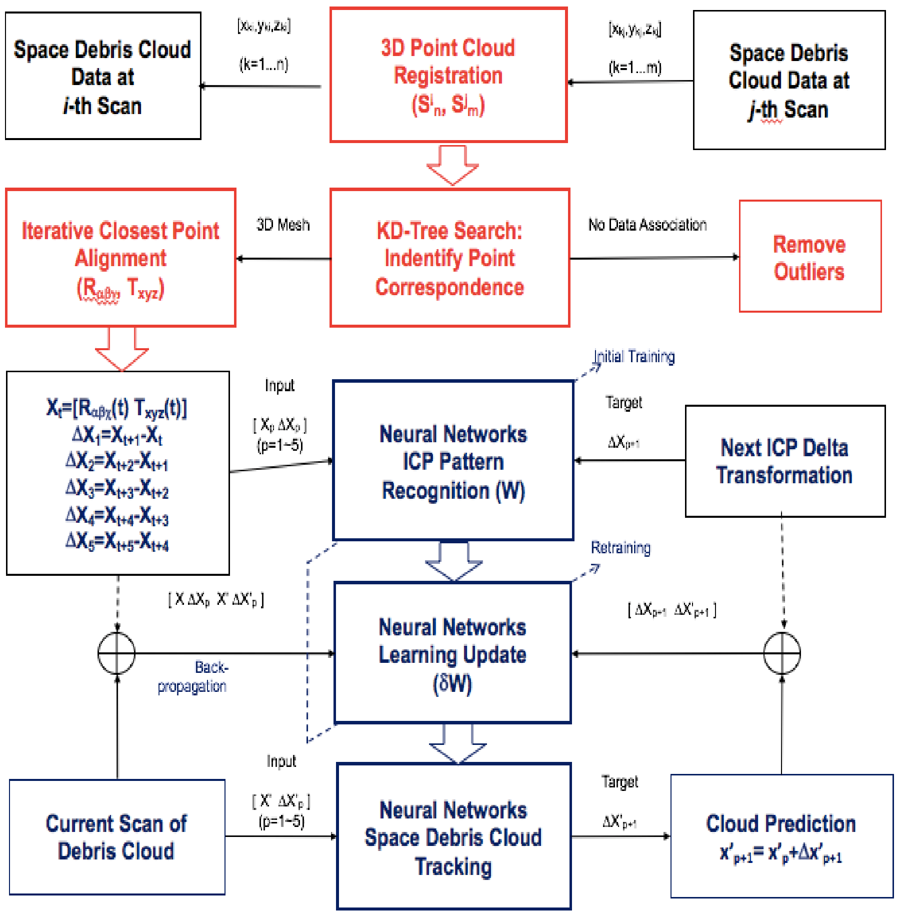
*where **a** is roll angle about x-axis, **b** a pitch angle about y-axis, and **g** a yaw angle about z-axis*

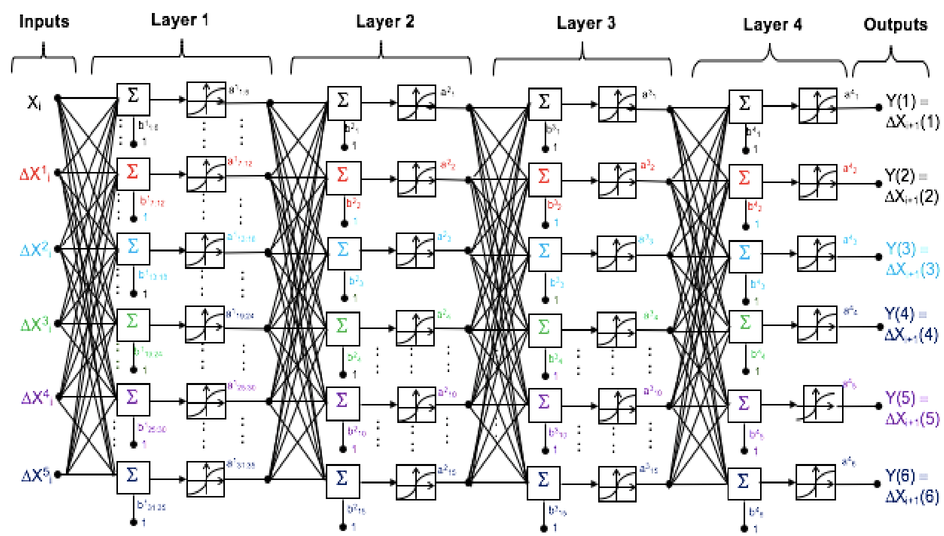
- Ultimately, ICP algorithm decides the congruence of different geometric representations and estimates motion and rotation between two point clouds where correspondences are not known.

ICP Approach to Mapping Space Debris Clouds



Trajectory Prediction ANN for Space Debris Cloud Tracking



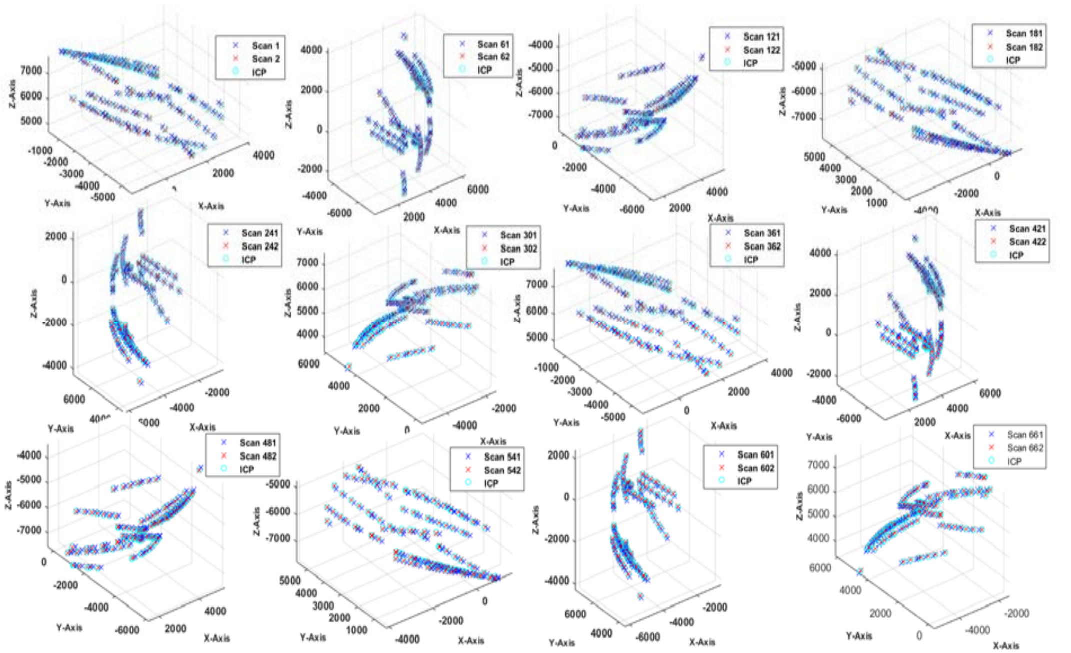


$X_i=[R_{\alpha\beta\chi}(i)\ T_{xyz}(i)]$, $\Delta X^1_i=X_{i+1}-X_i$, $\Delta X^2_i=X_{i+2}-X_{i+1}$, $\Delta X^3_i=X_{i+3}-X_{i+2}$, $\Delta X^4_i=X_{i+4}-X_{i+3}$, $\Delta X^5_i=X_{i+5}-X_{i+4}$

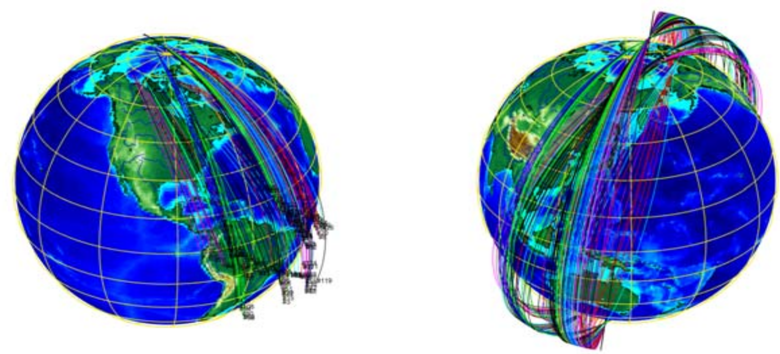
ANN	Input Layer	Hidden Layers/ Neurons	Output Layer	Training Rate	Weighting Momentum	MSE	Cloud Samples
Trajectory Prediction	36	2 / 15	6	0.4	0.3	1.0E-6	128

ICP Kinematic Patterns Applied to ANN

128 Space Debris Cloud ICP Scans Over 720-degree per 1-degree True Anomaly



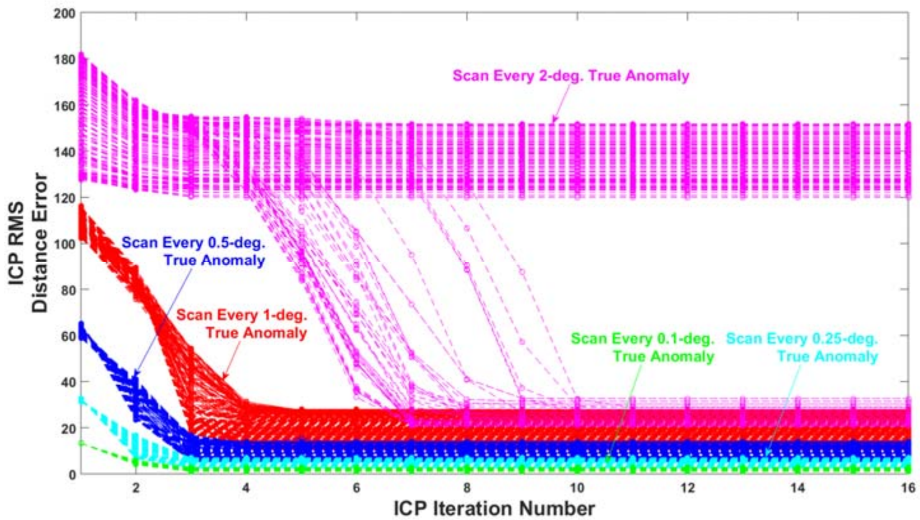
128 Two-Line Element (TLE) Data of Space Debris Samples for ICP Registration



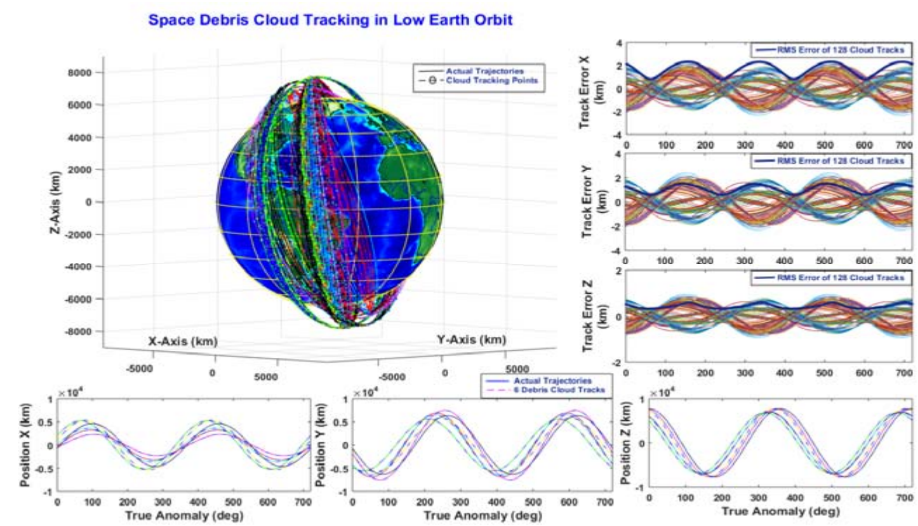
- Iterative Closest Point (ICP) features, which include scanning, meshing, and point correspondence via KD-tree search along with ICP alignment via the Singular-Value Decomposition (SVD) algorithm were executed for 128 samples of space debris clouds at an incremental true anomaly as scanned from 0 to 720 degrees.
- Artificial Neural Networks system will predict future changes of ICP kinematic patterns for space debris cloud tracking.

Simulation Results: Sensitivity Analyses of ICP

128 Cloud ICP RMS Distance Error vs. Scan Angle in True Anomaly



Space Debris Cloud ANN Tracking Using ICP Kinematic Patterns



- ICP kinematic patterns provided by the angles of $\mathbf{R}_{abg}$ and the displacements of $\mathbf{T}_{xyz}$ as well as five successive changes of them in six cloud scans are applied to the Trajectory Prediction Artificial Neural Network (ANN) for space debris cloud tracking over 720 degrees.
- ANN's predictions for future ICP kinematic patterns of the space debris cloud were compared with TLE real data of the space debris samples to obtain tracking errors.
- Average total tracking error for the space debris cloud group is 0.87 km (80 arcseconds) while the average tracking error for an individual space debris sample is less than 0.1 km (10-15 arcseconds).

Conclusion and Discussion

- ANN-based Orbital Recognition System is validated to execute accurate target detection and precision trajectory prediction functions using space-debris samples in terms of five Keplerian elements that present geometric orbital patterns
- Two ANNs integrated to work as an outer-space GPS for space debris tracking
  - Both ANNs trained using backpropagation method and retrained by learning new and/or corrected samples, if available, to both subsystems
- Sensitivity of Target Detection ANN was analyzed to clarify the bounds of the orbital variations for the desired prediction accuracy
- Simulation of Trajectory Prediction ANN for space debris shows that ANN system can interpolate the changes of orbital patterns in the waypoints that were never trained before
- Tracking errors for Trajectory Prediction ANN are smaller than those of conventional tracking methods
- Successful experimental applications of an ANN-based Orbital Recognition System confirm theoretical approach that pattern recognition ANNs can act as an accurate and effective space surveillance system for real space debris tracking

Future Works

- This research can be applied to any moving object with an elliptical orbit
  - e.g. satellites, space cargo, deep space planets, land drones
- Research will be continued by utilizing deep-learning algorithms to auto-encode orbital patterns of space debris samples without supervision
- Properties of space debris (size and mass) can be added as additional patterns to expedite pattern recognition of space debris
- Research will be extended to track space debris as clouds using point-cloud registration technique and interactive-closest-point algorithm in conjunction with the Keplerian state transition matrix for multi-orbit space debris cloud tracking

References

[1] B. G. Cour-Palais and D. J. Kessler, “Collision Frequency of Artificial Satellites: The Creation of a Debris Belt,” *Journal of Geophysical Research: Space Physics.*, June 1, 1978, vol. 83, pp. 2637-2646.

[2] D. J. Kessler, N. L. Johnson, J. –C. Liou, and M. Matney, “The Kessler Syndrome: Implications to Future Space Operations,” *Advances in Astronautical Sciences*, Feb. 6-10, 2010, vol. 137, pp. 47-61.

[3] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *Transactions of the ASME Journal of Basic Engineering*, Vol. 82, Series D, 1960, pp. 35-45

[4] D. A. Vallado and S. S. Carter, “Accurate Orbit Determination from Short-Arc Dense Observational Data,” *Journal of the Astronautical Sciences*, April 1998, vol. 46, pp. 195-213.

[5] O’Keefe et. al., “The Hippocampus As a Cognitive Map,” vol. 3, Oxford: Clarendon Press, 1978.

[6] N. Y. Xiao, “Using the Modified Back-Propagation Algorithm To Perform Automated Downlink Analysis,” *Department of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology*, June 1, 1996, pp. 23-31.

[7] W.H. Goodyear, “Completely General Closed-Form Solution for Coordinates and Partial Derivatives of the Two-Body Problem,” *The Astronomical Journal,* April 1965, vol. 70, no. 3, pp. 189-192.

[8] P. J. Werbos, “Backpropagation Through Time: What It Does and How To Do It,” *Proceedings of IEEE,* Oct. 1990, vol. 78, no. 10, pp. 1550-1554.

[9] “Competition Entrant”, “Orbital Recognition System for Space Debris Tracking Using Artificial Neural Networks — A Journey from Inner-Brain GPS to Outer Space GPS,” accepted for publication in *Journal of Emerging Investigators*, July 2016.

[10] P. Payeur, H. Le-Huy, and C. M. Gosselin, “Trajectory Prediction for Moving Objects Using Artificial Neural Networks,” *IEEE Transactions on Industrial Electronics,* April 1995, vol. 42 no. 2, pp. 147-150.

[11] Curtis, Howard D., *Orbital Mechanics for Engineering Students*, Elsevier Academic Press, Burlington, MA, 2005.

[12] G. R. Curry, *Radar System Performance Modeling*, Artech House, November 2004, pp. 165-193.
P.Vincent et. al., “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion,” *Journal of Machine Learning Research*, vol. 11, pp. 3371-3408, December 2010.

[13] P.J. Besl and N.D. McKay, “A Method for Registration of 3D Shapes,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239-256, February 1992.