GEOMAGNETIC MATCHING FOR NAVIGATION BASED ON NEURAL NETWORKS
: 뉴럴 네트워크 기반의 지구자기장 정합 항법

Presenter: Donghun Kim
Advisor: Hyochoong Bang

2019.05.21

KAIST

ASCL
Aerospace Systems & Control Lab.

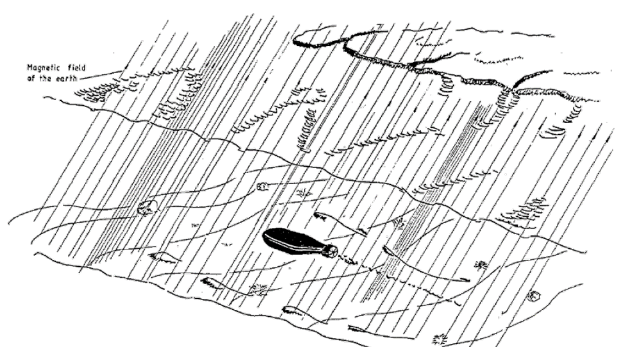
Table of contents

Opening

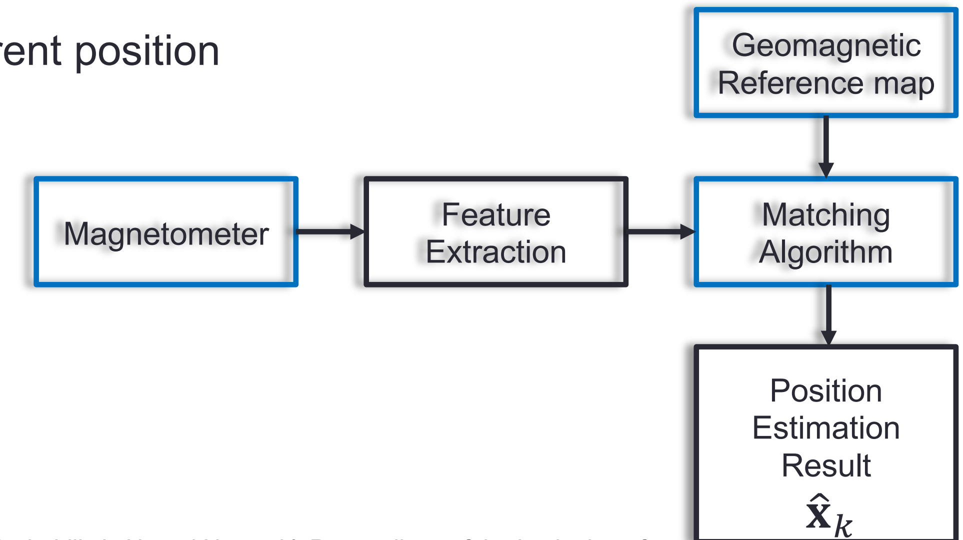
- **Introduction**
  - Geomagnetic matching for navigation
  - Comparison of navigation methods
  - Geomagnetic matching algorithms
  - Neural networks approach
  - Motivations
  - Objectives

Introduction

- **Geomagnetic matching for navigation**
  - ***Position estimation using the Earth's magnetic field information***
    - Find a position having geomagnetic values that corresponding to geomagnetic measurements on the geomagnetic reference map [1]
  - Definition of navigation
    - Moving from one place to another or looking for the route required for such move [2]
      - Important to know the current position



Magnetic terrain navigation [3]



[1] Zhou, J., Liu, Y., Ge, Z.: 'Geomagnetic Matching Algorithm Based on Probabilistic Neural Network'. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2010, 225, (1), pp. 120–126.
[2] Anderson, E.W.: 'The Principles of Navigation', Hollis and Carter, 1966.
[3] Tyrén, C.: 'Magnetic Terrain Navigation'. Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology, Durham, NH, USA, June 1987, pp. 245–256.

Introduction

- **Comparison of navigation methods**

Methods	Geophysics	Sensors	Reference Map	Available Area	Application System	Characteristics
Geomagnetic Aided Navigation (GAN)	Earth magnetic field	Proton precession/ flux gate magnetometer	WDMAM(1) EMAG 2(2) (=2 arcmin)	Airborne Maritime Land Indoor	MAGCOM(6)	-. ***Over 97% on the Earth*** -. ***Less dependent on weather*** -. ***Miniaturization sensor*** -. Low map resolution -. Magnetic interference
Terrain Reference Navigation (TRN)	Terrain elevation	Barometer, Altimeter, LRF, LiDAR	DTED(3) LV 2 (1 arc-sec)	Airborne	TERCOM(7) SITAN(8) TERPROM(9) PTAN(10)	-. Proven application systems -. Sensitive terrain characteristics -. Weather, altitude uncertainty -. Higher altitude, higher power
Bathymetry Referenced Navigation (BRN)	Under water terrain elevation	Pressure altimeter, Echosounder, Sonar, DVL	GEBCO(4) (30 arc-sec)	Maritime	TRIN(11)	-. Specialized in maritime application -. Tidal uncertainty
Gravity Gradient Referenced Navigation	Gravitational acceleration anomaly	Gravity gradiometer	WGM(5)	Airborne Maritime Land	-	-. All over the Earth -. Expensive sensor -. Not miniaturized

(1)World digital magnetic anomaly map, (2)Earth magnetic anomaly grid, (3)Digital terrain elevation database, (4)General bathymetric chart of the oceans, (5)World gravity model, (6) Magnetic COntour Matching , (7)Terrain COntour Matching, (8)Sandia Terrain Aided Navigation, (9)TERrain P ROfile Matching, (10)Precise Terrain Aided Navigation, , (11)Terrain referenced integrated navigation

Introduction

- **Geomagnetic matching algorithms**

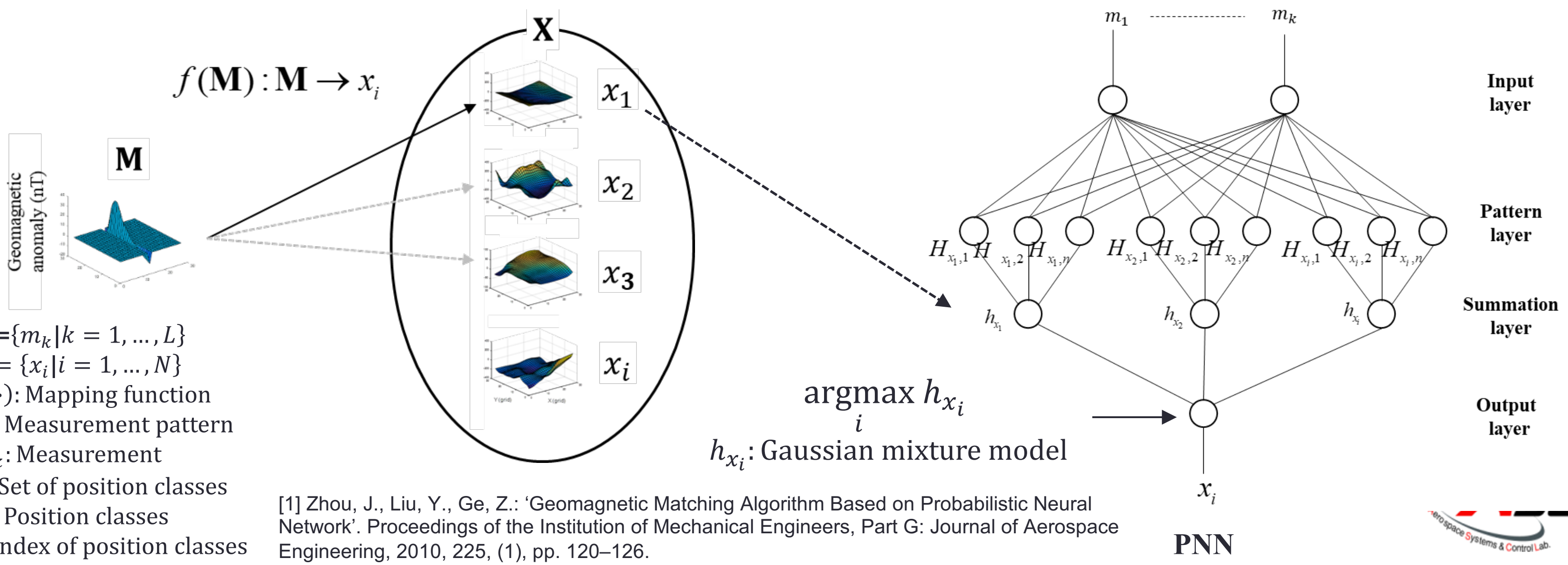
Class	Algorithms	Specification	Characteristics	Geomagnetic	Remark
Correlation	MSD(1)	**Correlate vector to reference map:** Measurement vector Batch processing	Initialization sensitivity **Not allowed maneuver** **Limited update period**	Xie et al.: sea [5]	TERCOM MAGCOM
	ICCP(2)	**Correlate vector to trust track lines:** Measurement vector Iterative processing	Initialization sensitivity **Linear and uniform motion**	Zhang et al.: sea [6]	*Vector ICCP/ Chen et al.: land [7]*
Filter	EKF(3)	**Estimate error state or position:** Single measurement Prediction/measurement update	On-line processing **Map function linearization error**	Feng et al.: sea [8] Mu et al.: land [9]	SITAN TERPROM (MSD + EKF)
	PF(4)	**Estimate position:** Single measurement Propagate/update/resampling	Easy to implement **High computational load** Degeneracy/impoverishment	Canciani et al.: air [10] Kauffman et al.: land [11]	Gustafsson et al. [12]
Neural Networks (NNs)	PNN(5)	**Classify the pattern to position:** Classification-based approach Measurement vector Modeling sum of conditional pdf	Linear and uniform motion **Not applicable to new area** **High NNs complexity for navigation area**	Zhou et al.: air [1]	

(1)Minimum squared difference, (2)Iterative closest contour matching, (3)Extended Kalman filter, (4)Particle filter, (5)Probabilistic neural network

:

Introduction

- **Neural networks approach**
  - Probabilistic neural network (PNN) [1]
    - Classification-based geomagnetic matching algorithm
      - ***Mapping input measurement patterns to specific position classes***
      - Based on Gaussian mixture model with simple in/output structure
    - Characteristics of PNN
      - ***The navigation area is confined to the trained area***
        - ***The complexity of PNN increases as the position increases***
        - The summation layers must contain all position information to classify



$\mathbf{M}=\{m_k | k = 1, \ldots, L\}$
$\mathbf{X} = \{x_i | i = 1, \ldots, N\}$
$f(\cdot)$: Mapping function
$\mathbf{M}$: Measurement pattern
m_k: Measurement
$\mathbf{X}$: Set of position classes
x_i: Position classes
i: Index of position classes

[1] Zhou, J., Liu, Y., Ge, Z.: 'Geomagnetic Matching Algorithm Based on Probabilistic Neural Network'. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2010, 225, (1), pp. 120–126.

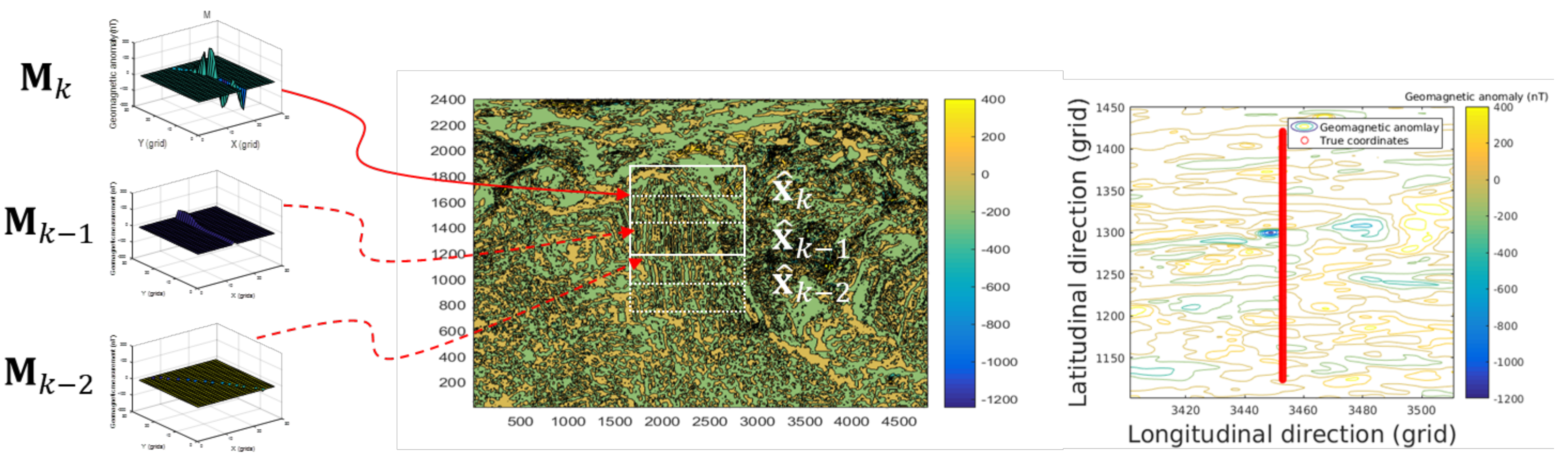
Introduction

- **Motivations**
  - Geomagnetic has the advantages of wide applicability [10]
    - ***Available over 97% on the Earth***
    - ***Less dependent on weather, day and night, and seasonal changes***
    - Miniaturization sensors
  - NNs based approach is still a pioneering area with challenging issues
    - ***Defining NNs structure applicable to the new area without training***
      - The complexity of NNs is independents on the navigation area
      - Extract information needed for position estimation from only inputs
    - ***Generating datasets and defining a loss function to train NNs***
      - NNs need big data and loss function to train
      - Only a small number of training samples are available on a position

[10] Canciani, A., C., Raquet, J.: 'Airborne Magnetic Anomaly Navigation', IEEE Trans. Aerosp. Electron. Syst., 2017, 53, (1), pp. 67–80.

Introduction

- **Objectives**
  - Propose a new approach to geomagnetic matching for navigation based on NNs combined with the correlation method
    - ***Enlarging the navigation area to the area where geomagnetic information is available***
      - Removing dependency between navigation area and NNs complexity
      - Possible to estimate positions using measurements coming from the new area



Geomagnetic measurement pattern **Reference geomagnetic map** **Results of position matching**

$\mathbf{M}_k$: Measurement pattern
$\hat{\mathbf{x}}_k$: Estimated position vector

Methodologies

- **Proposed scheme**
  - Distance-based approach
  - Block diagram
- **Abstraction step**
  - Measurement pattern
  - Small search maps
  - Two symmetric convolutional neural networks (CNNs)
  - Training two symmetric CNNs
- **Refinement step**
  - Normalized cross-correlation (NCC)
  - Continuous position estimation

Proposed scheme

- **Distance-based approach**
  - ***Abstraction step*** based on CNNs
    - ***Retrieving a candidate region similar to the measurement pattern***
      - Calculating ***similarity metric*** between measurement pattern and candidates
      - Selecting a candidate region with the minimum distance

Similarity metric

$$D_W(\mathbf{M},\mathbf{S}) = \left\| G_W(\mathbf{M}) - G_W(\mathbf{S}) \right\|_2$$

$\mathbf{M}$: Measurment pattern, $\mathbf{M} \in {}^2$

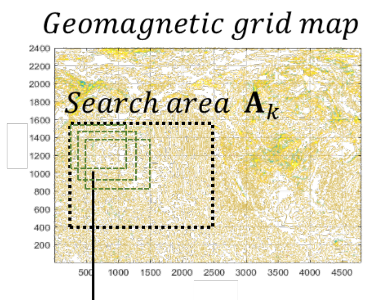
$\mathbf{S}$: Small search map, $\mathbf{S} \in {}^2$

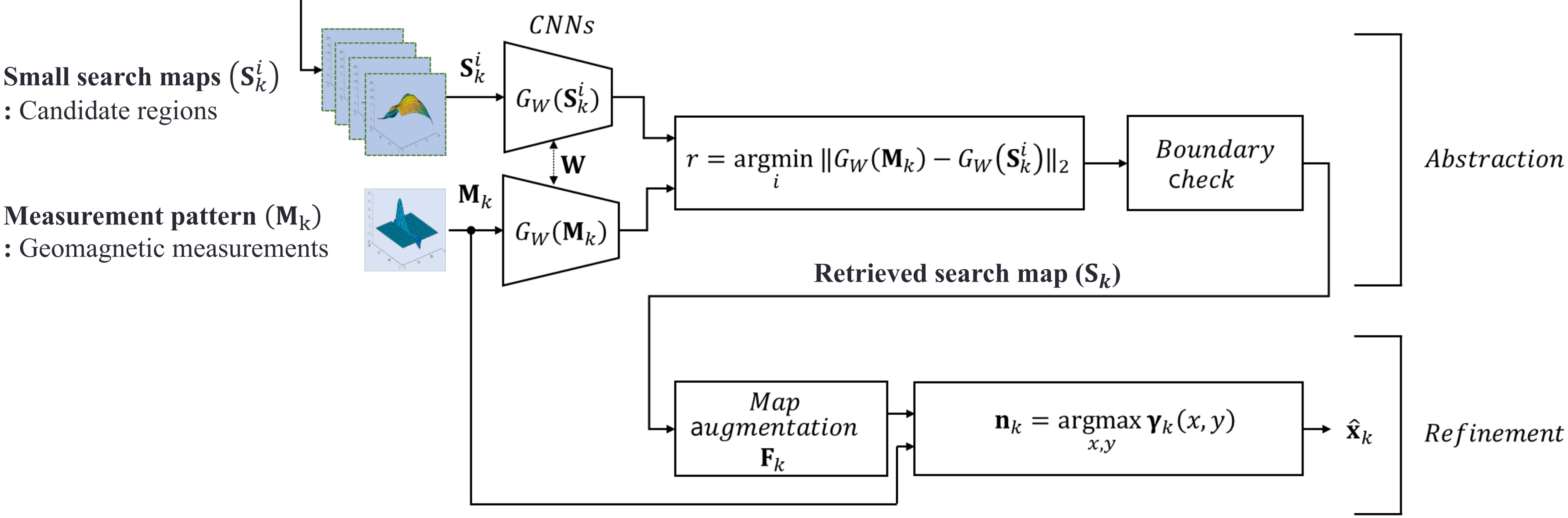
$G_W(\cdot)$: CNNs

- ***Refinement step*** based on correlation method
  - ***Finding matching position using normalized cross-correlation (NCC)***
    - Calculating the correlation in the retrieved candidate region
    - Selecting the maximum correlation position in the retrieved candidate region

Proposed scheme

- **Block diagram**





Definition of map

- A set of geomagnetic intensity values mapped on 2-D with the grid resolution and specific size
- Represented in **bold italic character**
- Ex) Samll search maps: $\mathbf{S}_k^i$,
  Augmented map: $\mathbf{F}_k$

Abstraction step

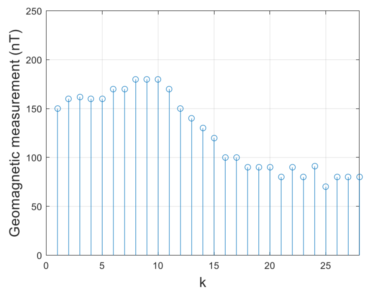
- **Measurement pattern ($\mathbf{M}_k$)**
  - Geomagnetic measurements ***mapped onto the spatially quantized 2-D plane*** with grid resolution (δ) using grid adjustment

Measurement pattern

$\mathbf{M}_k \in \;^2$

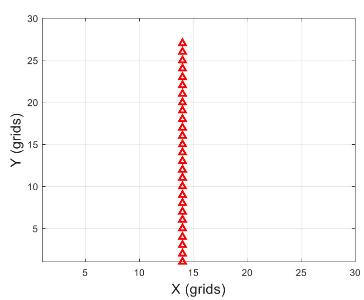
size of $\mathbf{M}_k$: $n \times n$ grids with grid resolution $\delta = g_k$

Geomagnetic measurements



$$\mathbf{m}_k = \{z_{k-j}\}_{j=0}^{N}$$

Position



$$\mathbf{X}_k = \{\mathbf{x}_{k-j}\}_{j=0}^{N}$$

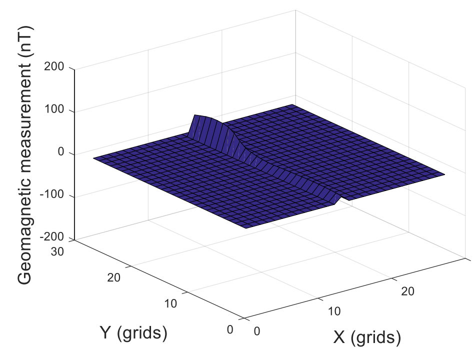
Calculate the grid resolution
$g_k = \Delta x_{\min} \times SF$

Calculate the number of measurements
$N : l_{\max} \leq n \times g_k$

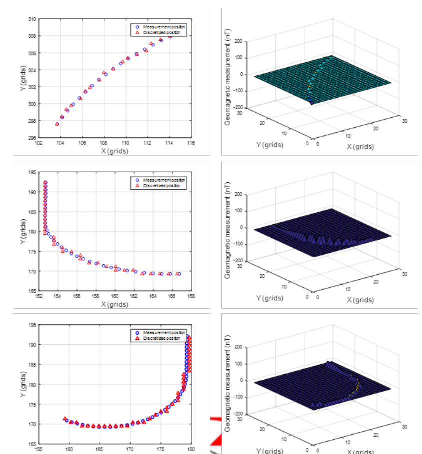
Assign the measurements to discrete grid positions
$\mathbf{M}_k(\mathbf{x}'_{k-j}) \leftarrow m_k(j)$

Grid adjustment

$\mathbf{m}_k$: Set of measurements z_k
z_k : Geomagnetic measurment
$\mathbf{X}_k$: Set of position vectors
$\mathbf{x}_k$: Position vector corresponding to z_k
g_k : Grid resoultion at k
$\Delta x_{\min}$: Minimum grid size
SF : Scalefactor, $0 < SF \leq 1$
$l_{\max}$: Maximum length
N : Number of measuremen

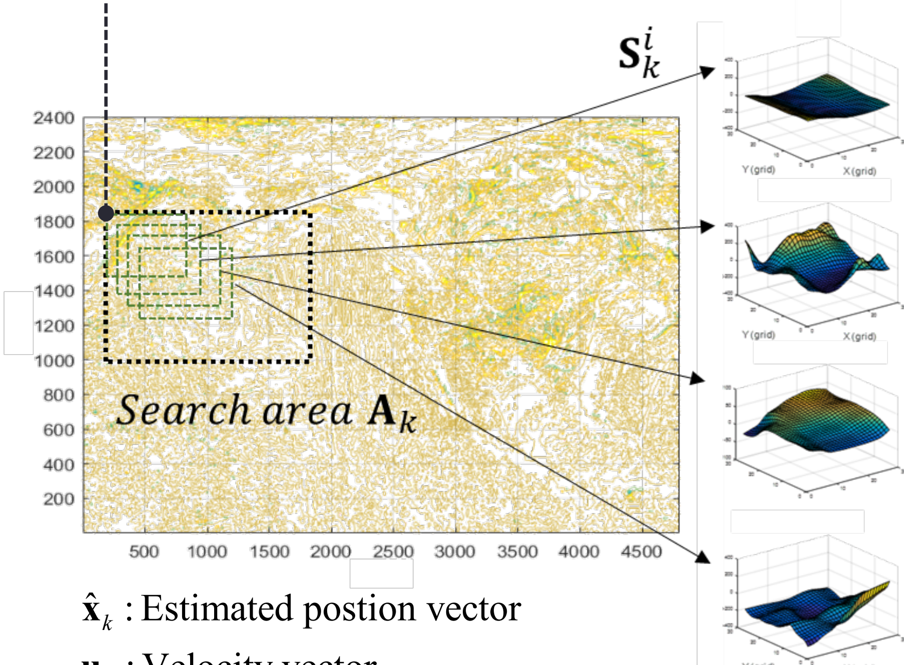


Measurement pattern $\mathbf{M}_k$



Abstraction step

- **Small search maps (S_k^i)**
  - ***Candidate maps to be matched with a measurement pattern*** $(\mathbf{M}_k)$
    - Subsets of geomagnetic values in a search area $(\mathbf{A}_k)$
      - $\mathbf{A}_k$ is defined at origin position $(\mathbf{a}_k)$ having $m \times m$ size with $\delta = g_k$, where $m > n$
      - $\mathbf{S}_k^i$ is defined at all origin positions of $\mathbf{b}_k^i$ having $n \times n$ grids size with $\delta = g_k$



$\hat{\mathbf{x}}_k$: Estimated postion vector

$\mathbf{u}_k$: Velocity vector

Origin position of $\mathbf{A}_k$

$$\mathbf{a}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{u}_k \cdot dT - \frac{g_k}{2}[m, m]^T$$

Small search maps ($\mathbf{S}_k^i$)

$\mathbf{S}_k^i \subset \mathbf{A}_k$

$[0,0]^T \leq \mathbf{b}_k^i \leq \left[(m-n)-1, (m-n)-1\right]^T$

$\mathbf{b}_k^i$: Relative origin postion of $\mathbf{S}_k^i$

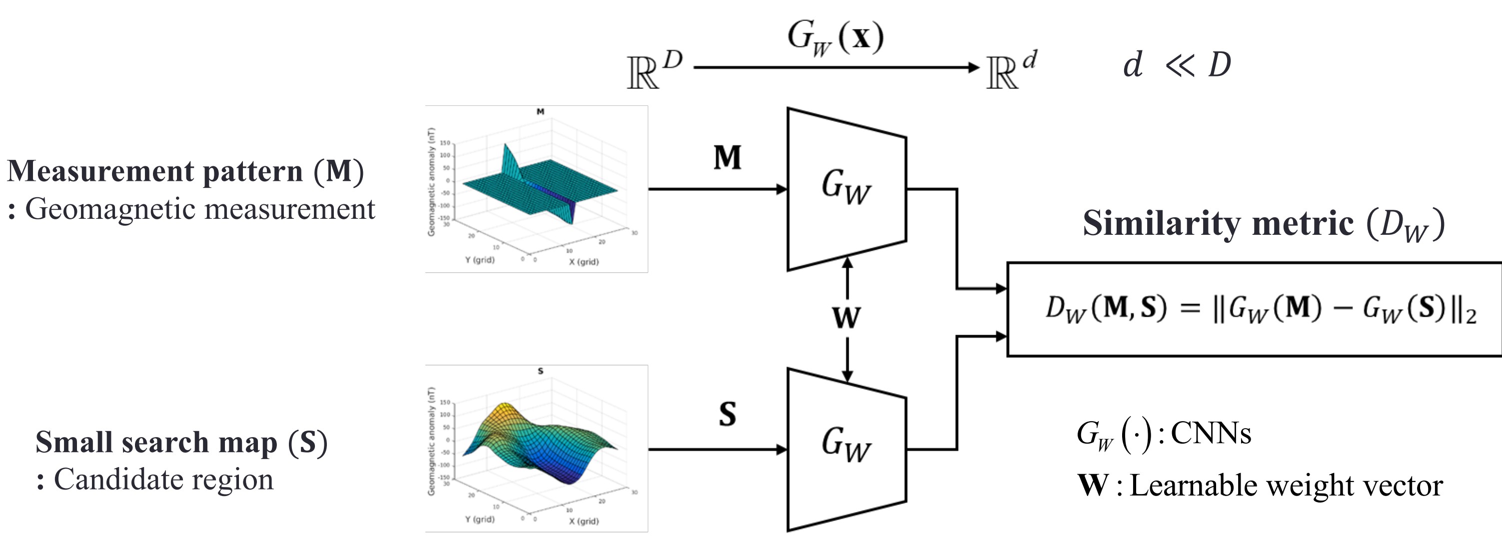
$i = 1, \ldots, (m-n)^2$

Size of $\mathbf{S}_k^i$: $n \times n$ grids with δ

Size of $\mathbf{A}_k$: $m \times m$ grids with δ

Abstraction step

- **Two symmetric CNNs**
  - Siamese neural network architecture [11]
  - ***CNNs transform complex data into simple feature data***
    - ***Calculating the similarity metric using the feature data***
      - If **M** and **S** are matched data pair, the similarity metric is shorter
      - If **M** and **S** are mismatched data pair, the similarity metric is longer
  - Advantages
    - ***Fixed complexity of NNs regardless of the size of the navigation area***
    - ***Applicable to all areas where inputs needed to calculate*** $\mathrm{D_W}$ ***are provided***



[11] Bromley, J., Guyon, I., LeCun, Y., Sackinger, E., Shah. R.: ‘Signature Verification using a Siamese Time Delay Neural Network’. Proceedings of the 6th International Conference on Neural Information Processing Systems, Denver, CO, USA, December 1993, pp. 737–744.

$(Y, \mathbf{M}, \mathbf{S})$: Matched data pair
$(Y, \mathbf{M}, \mathbf{S}')$: Mismatched data pair
Y : Binary label
$\mathbf{M}$: Mesurement pattern
$\mathbf{S}$: Search map matched to $\mathbf{M}$
$\mathbf{S}'$: Search map mismatched to $\mathbf{M}$
η : Learning rate

Abstraction step

- **Training two symmetric CNNs**
  - Finding **W** to calculate similarity metric using feature data
    - Loss (L) minimization by updating **W** recursively with training datasets

Binary label $Y = \begin{cases} 0, & \text{if matched data pair} \\ 1, & \text{if mismatched data pair} \end{cases}$

Matched Dataset $\mathbf{D}_M$

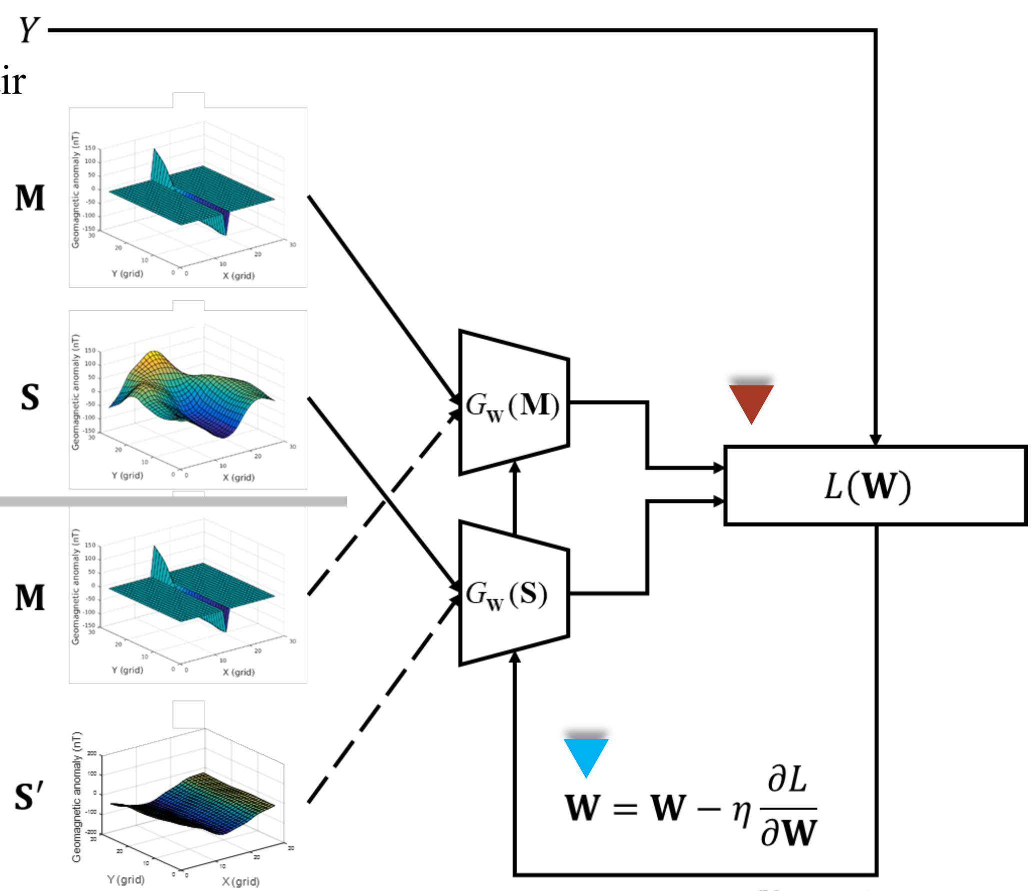
$$\mathbf{D}_M = \left\{ \left(Y, \mathbf{M}, \mathbf{S} \right)^i \right\}_{i=1}^{N}$$

Mismatched Dataset $\mathbf{D}_S$

$$\mathbf{D}_S = \left\{ \left(Y, \mathbf{M}, \mathbf{S}' \right)^i \right\}_{i=1}^{N}$$

$i = 1, \ldots, N$

i : index of samples

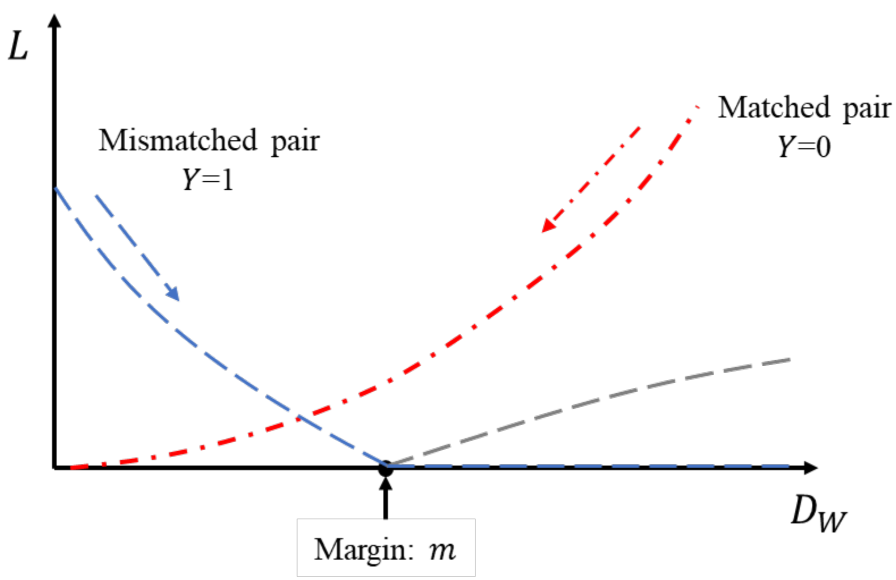


Abstraction step

Similarity metric

$$D_W(\mathbf{M},\mathbf{S}) = \left\| G_W(\mathbf{M}) - G_W(\mathbf{S}) \right\|_2$$

- **Training two symmetric CNNs**
  - Loss function
    - Defining ***contrastive loss function*** to train two symmetric CNNs
      - ***Minimize D_W, if matched data pair from the matched dataset***
      - ***Maximize D_W, if mismatched data pair from the mismatched dataset***



Contrastive loss function

$$l\left(\mathbf{W},\left(Y,\mathbf{M},\mathbf{S}\right)\right) =$$

$$\frac{1}{2}\times(1-Y)\times \underbrace{L_M\left(D_W(\mathbf{M},\mathbf{S})\right)^2}_{\text{Matched pair:}} + \frac{1}{2}\times Y \times \underbrace{L_S\left(D_W(\mathbf{M},\mathbf{S})\right)^2}_{\text{Mismatched pair:}}$$

$$L_M\left(D_W(\mathbf{M},\mathbf{S})\right) = \left\| G_W(\mathbf{M}) - G_W(\mathbf{S}) \right\|_2$$

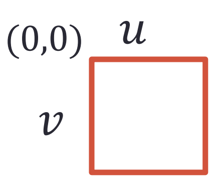
$$L_S\left(D_W(\mathbf{M},\mathbf{S})\right) = \max\left(0, \left(m - D_W(\mathbf{M},\mathbf{S})\right)\right)$$

Loss function

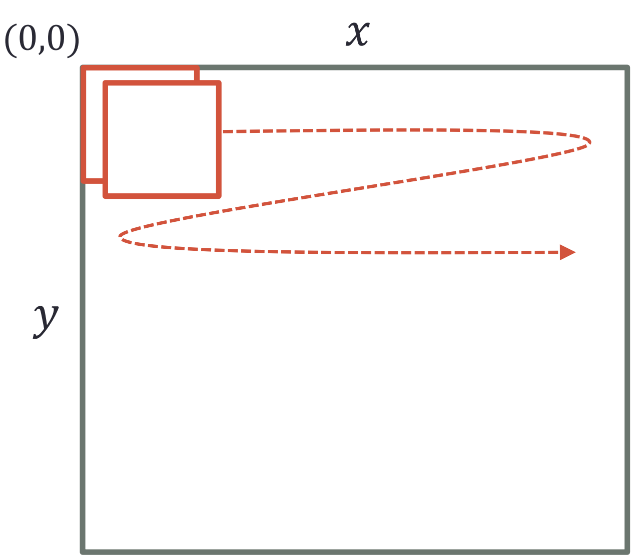
$$L(\mathbf{W}) = \sum_{i=1}^{p} l\left(\mathbf{W},\left(Y,\mathbf{M},\mathbf{S}\right)^i\right) \quad p\text{: Batch processing size}$$

Refinement step

- **Normalized cross-correlation**
  - Calculating the spatial correlation
    - Between measurement pattern ($\mathbf{M}_k$) and augmented map ($\mathbf{F}_k$)
    - Selecting the largest correlation value for the final matching position



Measurement pattern $(\mathbf{M}_k)$



Augmented map $(\mathbf{F}_k)$

NCC

$$\gamma_k(x,y) = \frac{\sum_{u=0}^{n-1}\sum_{v=0}^{n-1} F\left((x+u, y+v)-\bar{F}\right)\times\left(M(u,v)-\bar{M}\right)}{\sigma_F \sigma_M}$$

Matching position

$$\mathbf{n}_k = \underset{x,y}{\arg\max}\, \gamma_k(x,y)$$

$\mathbf{n}_k$: relative matching position

$$\sigma_F = \sqrt{\sum_{u=0}^{n-1}\sum_{v=0}^{n-1}\left(F(x+u, y+v)-\bar{F}\right)^2}$$

$$\sigma_M = \sqrt{\sum_{u=0}^{n-1}\sum_{v=0}^{n-1}\left(M(u,v)-\bar{M}\right)^2}$$

σ_F : Variance of $\mathbf{F}_k$

σ_M : Variance of $\mathbf{M}_k$

$F(x,y)$: Returns a measurment at (x,y) in $\mathbf{F}_k$

$M(u,v)$: Returns a measurment at (u,v) in $\mathbf{M}_k$

$\bar{F}$: Mean of $\mathbf{F}_k$ overraped by $\mathbf{M}_k$

$\bar{M}$: Mean of $\mathbf{M}_k$

Refinement step

$\mathbf{a}_k$: Origin position of search area
$\mathbf{b}_k$: Origin position of retrieved search map
$\mathbf{c}_k$: Origin position of augmented map
$\mathbf{n}_k$: Relative position of maximum NCC

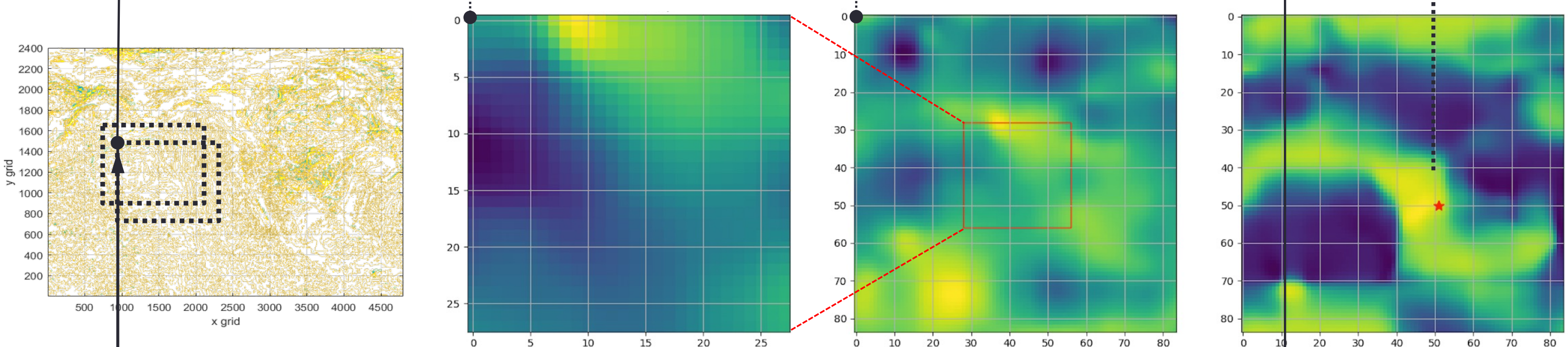
- **Continuous position estimation**
  - Done by recursively update origin positions of each step at k

Search area $(\mathbf{A}_k)$ **Retrieved search map** $(\mathbf{S}_k^{i=r})$ **Augmented map** $(\mathbf{F}_k)$ **NCC** $(\mathbf{n}_k)$

$$\mathbf{b}_k = \left(\mathbf{a}_k + \mathbf{b}_k^{i=r} \cdot g_k\right)$$

$$\mathbf{c}_k = \mathbf{b}_k - g_k \cdot [p-n, p-n]^T$$

$$\hat{\mathbf{x}}_k = \mathbf{c}_k + \mathbf{n}_k \cdot g_k$$



$$\mathbf{a}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{u}_k \cdot dT - \frac{\mathbf{g}_k}{2}[m, m]^T$$

Numerical simulations

- **Training two symmetric CNNs**
  - Training conditions
  - Training results
- **Numerical simulations**
  - Simulation conditions
  - Simulation parameters
  - Numerical simulation results
  - Analysis numerical simulations
  - Geomagnetic matching error compensation
  - Quantification of geomagnetic profile
  - Adaptive geomagnetic matching (AGM)
  - Summary of numerical simulation results

Training two symmetric CNNs

- **Training conditions**
  - ***5 convolutional layers + 1 fully connected layer***
  - Generated 3 datasets by adding zero-mean Gaussian noise
    - Each dataset has 30,000 data pairs

Training datasets

Random noise	Matched datasets	Mismatched dataset
0 nT	$\mathbf{D}_{M0}$	$\mathbf{D}_{S0}$
1.5 nT	$\mathbf{D}_{M1.5}$	$\mathbf{D}_{S1.5}$
3 nT	$\mathbf{D}_{M3}$	$\mathbf{D}_{S3}$

*nT: nano tesla ($1\ \mathrm{G} = 10^{-4}\ \mathrm{T}$)

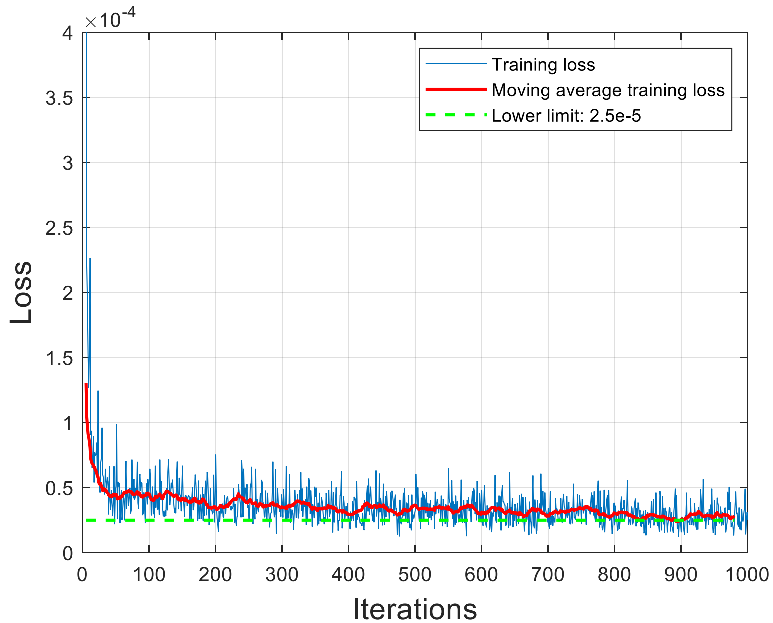
Training conditions

Parameters	Specification	Remarks
Target loss	2.5×10^{-5}	
Validation check error rate	$\leq 5\ \%$	Check after training finished
Margin (m) of loss fcn.	0.9	
Batch size	300	
Optimization	RMSProp	
System	Tensorflow + Pyhton	2,048 GPGPU cores

:

Training two symmetric CNNs

- **Training results**
  - Target loss achieve to be less than 2.5×10^{-5}
  - Validation check error rate achieve to be less than 5%



Result of the loss function
: Training iterations vs. loss function
(Iterations × 9,000)

Results of the validation check error rate

Validation datasets	V_{err}	Unit
D_0	3	[%]
$D_{1.5}$	4.9	
D_3	2.7	

:

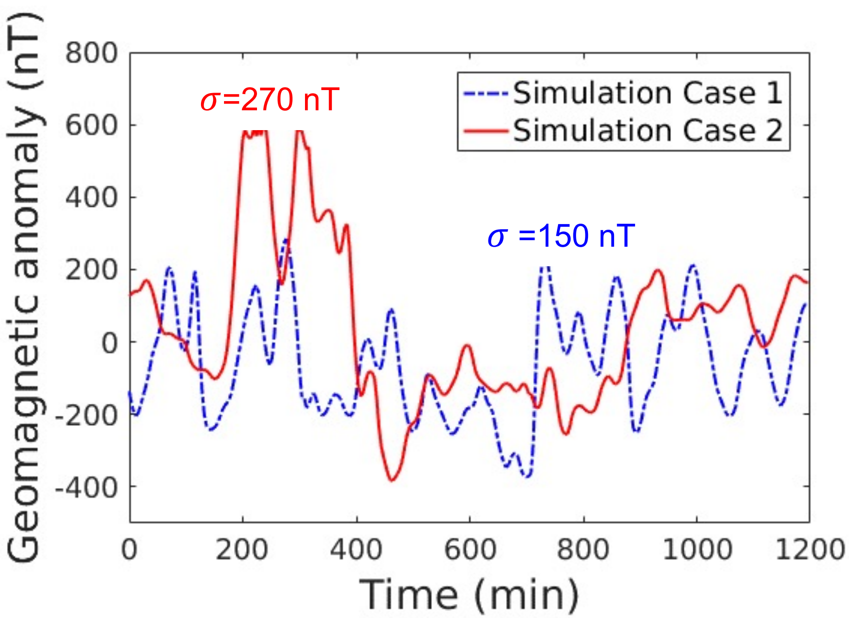
Validation check error rate

$$V_{err}[\%] = \frac{n_{total} - n_{match}}{n_{total}} \times 100$$

n_{total} : Number of total data
n_{match} : Number of matching

Numerical simulations

- **Simulation conditions**
  - Geomagnetic matching for navigation using ***2-D geomagnetic anomaly reference map*** on the ocean for vessel
    - Simulation Case 1
      - The best simulation case: even geomagnetic profile
    - Simulation Case 2
      - The worst simulation case: sharp and repeated less changing geomagnetic profile



Geomagnetic measurement profile

Dynamics, sampling period, and measurement noise

Conditions	Quantity	Unit
Vessel movement	30	knot
Sample period (k)	4	min
Measurement noise	≤ 3	nT

Numerical simulations

- **Simulation parameters**

Geomagnetic anomaly map

Category	Quantity	Unit
Grid resolution	2	arcmin
	3.7	km
Total number of grids	4800×2400	grid
Total map size	$17{,}779 \times 8{,}889$	km^2

Parameters for Abstraction step

Parameters	Quantity	Unit
Size of $\mathbf{A}_k$	56×56	grid
Size of $\mathbf{M}_k$	28×28	grid
Size of $\mathbf{S}_k^i$	28×28	grid
Boundary check range	16	grid/k

Parameters for Refinement step

Parameters	Quantity	Unit
Size of $\mathbf{F}_k$	31×31	grid

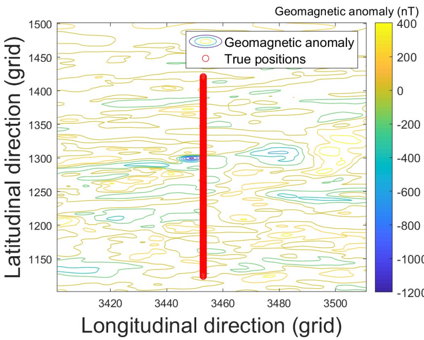
Numerical simulations

- **Numerical simulation results**

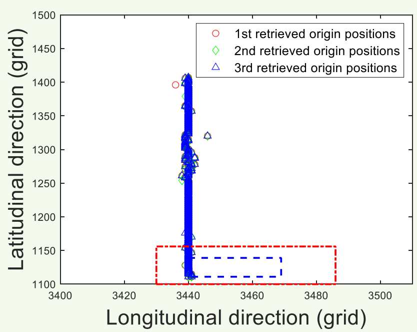
Simulation Case 1

Abstraction step

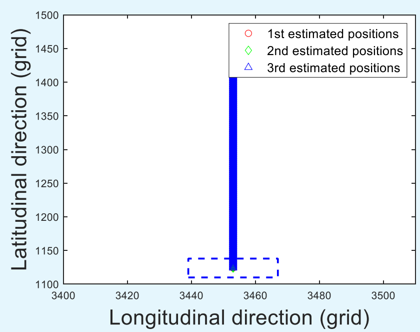
Refinement step



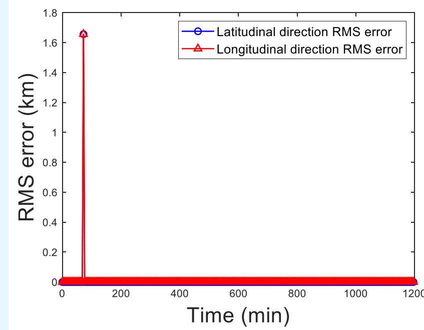
(a) True position



(b) Origin positions of retrieved search map

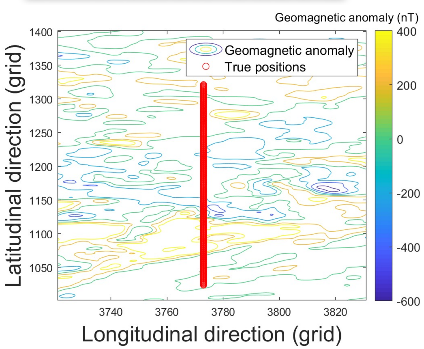


(c) Matching positions

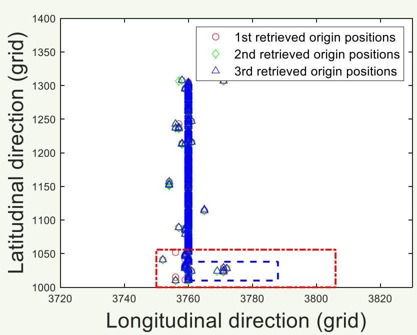


(d) RMS error

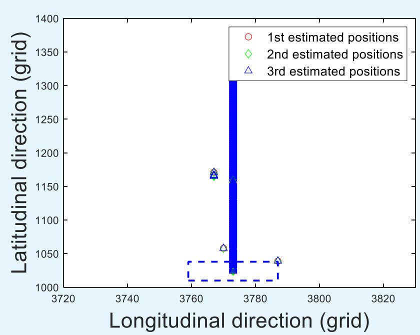
Simulation Case 2



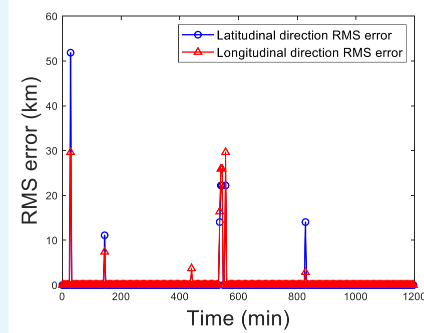
(a) True position



(b) Origin positions of retrieved search map



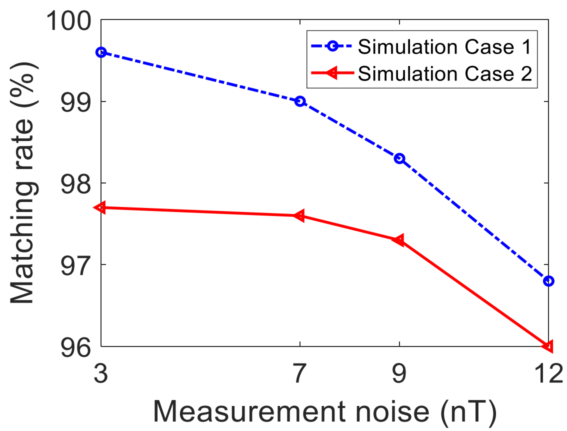
(c) Matching positions



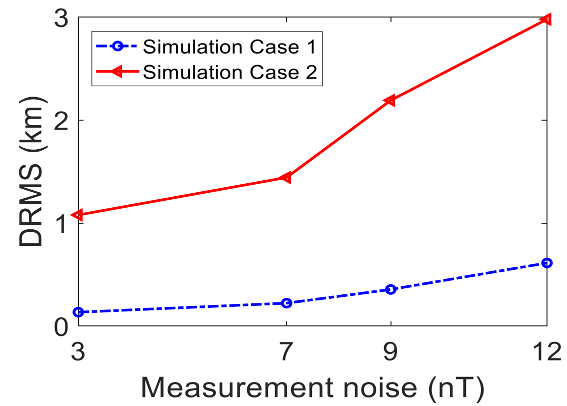
(d) RMS error

Numerical simulations

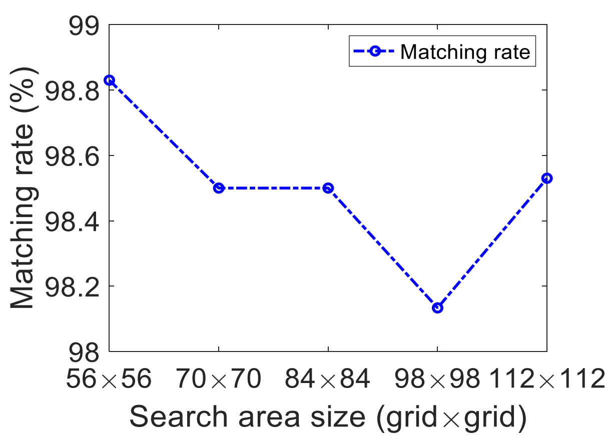
- **Analysis numerical simulations**



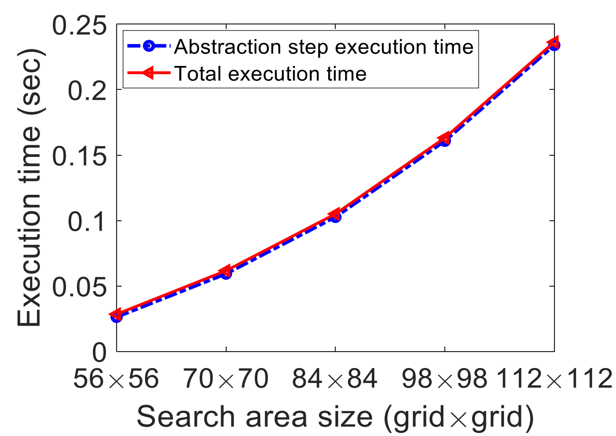
(a) Measurement noise vs. matching rate



(b) Measurement noise vs. DRMS



(c) Search area vs. matching rate

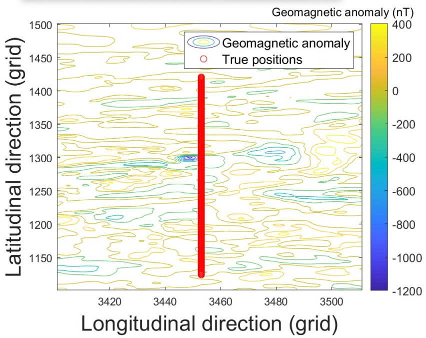


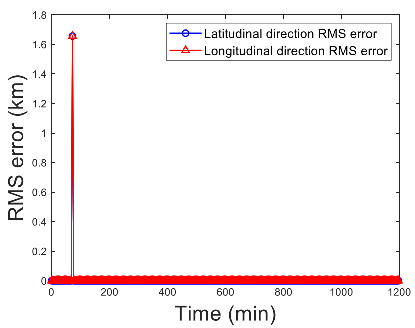
(d) Search area vs. computational load

Numerical simulations

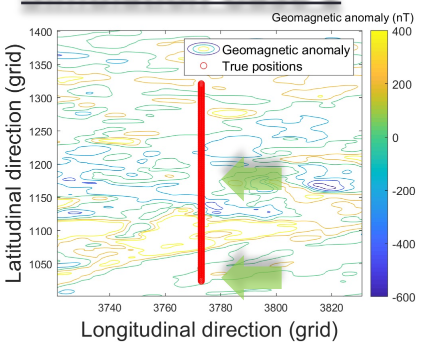
- **Geomagnetic matching error compensation**
  - Confirmed that the position errors increase if the geomagnetic profile is less change
    - In the case of TRN, position estimation errors increase when the terrain profile is flat or repeated [12]

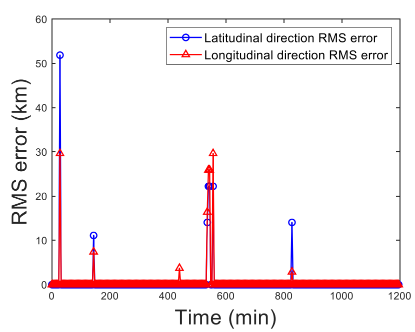
Simulation Case 1





Simulation Case 2





Two problems to compensate the errors:

- ***How to quantify the characteristics of the geomagnetic profile***
  : Direct/indirect methods
- ***How to compensate using quantified characteristics***
  : Adjust search area size

[12] Bergman, N.: 'Recursive Bayesian Estimation: Navigation and Tracking Applications'. PhD dissertation, Linköping University, 1999.

Numerical simulations

- **Quantification of geomagnetic profile**
  - Defining quantification methods for the geomagnetic profile

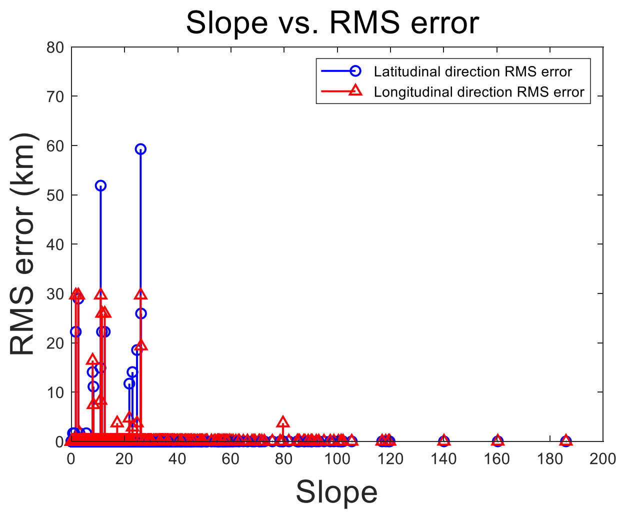
Method	Methods	Equations	Comments
Direct : Geomagnetic measurement	Slope	$s_k = \sqrt{(z_k - z_{k-1})^2}$	-. ***Assume no false alarm*** -. ***Timing margin required due to delay***
	Slope rate	$a_k = \sqrt{(s_k - s_{k-1})^2}$	
Indirect : 2-D search area	σ_T	$\sigma_T = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(z_i - \bar{z})^2}$, $\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z_i$	-. ***No delay occurred due to evaluate geomagnetic profile on 2-D plane*** -. ***High computational complexity required according to the size of the search area***
	σ_Z	$\sigma_Z = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N-1}(D_i - D)^2}$, $D_i = z_i - z_{i-1}$, $D = \frac{1}{N-1}\sum_{i=1}^{N-1} z_i$	

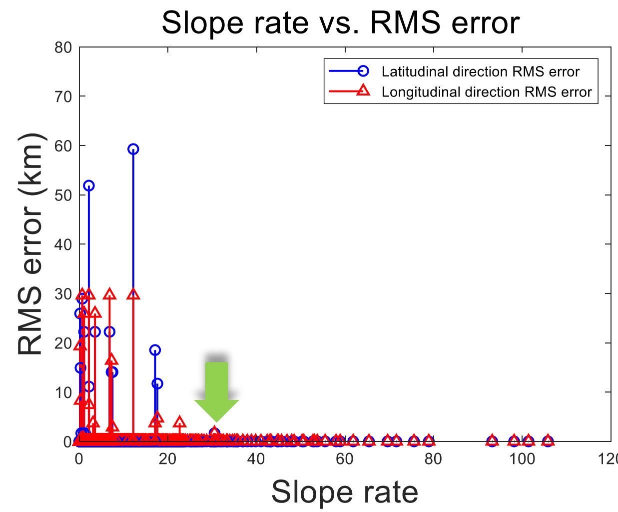
z_k: Geomagnetic measuremet

Numerical simulations

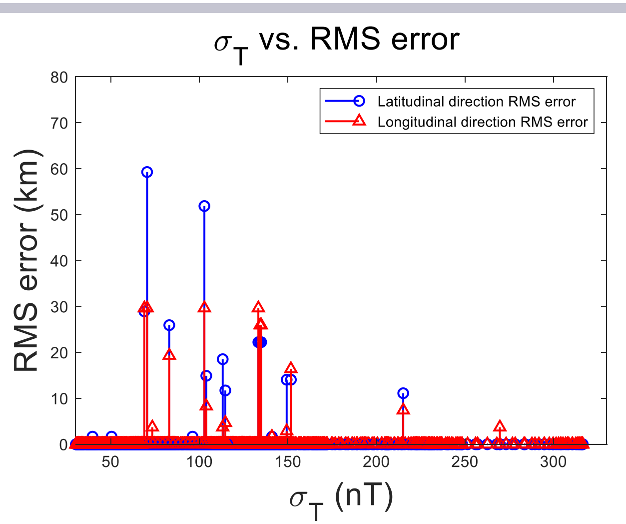
- **Quantification of geomagnetic profile**
  - Presenting quantification methods using 5 simulation cases

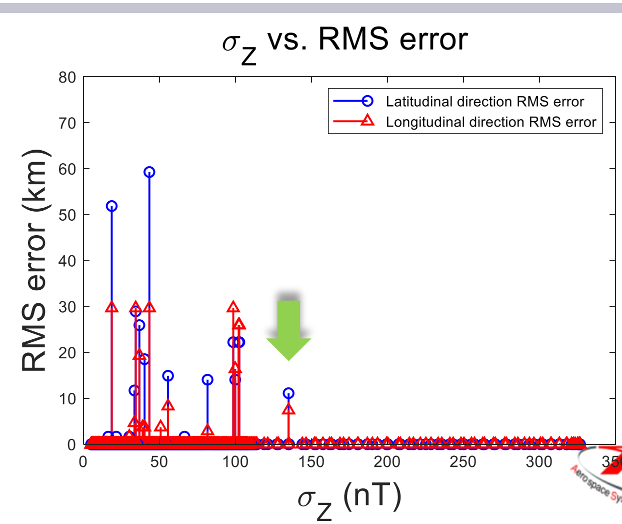
Direct Method





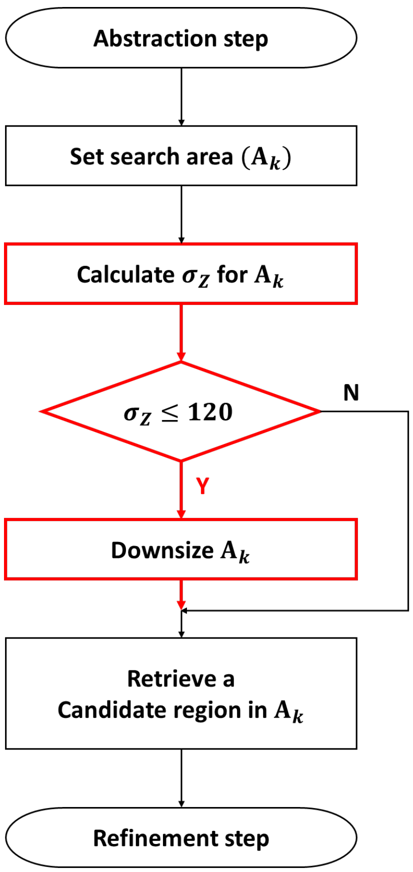
Indirect Method

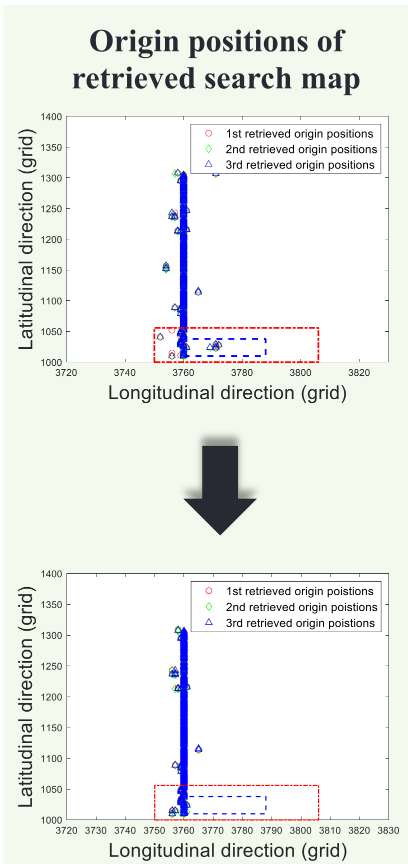


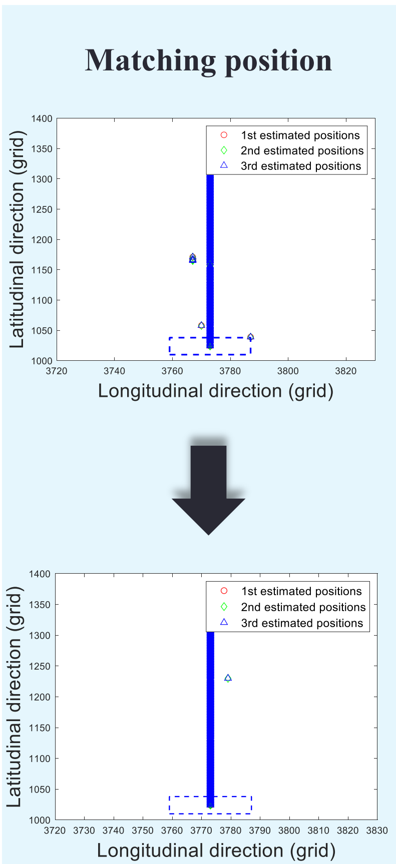


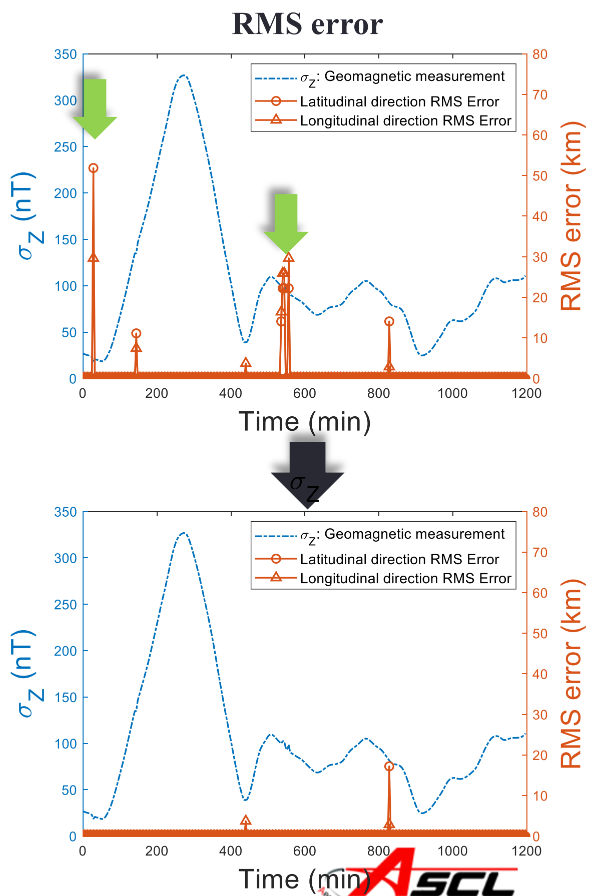
Numerical simulations

- **Adaptive geomagnetic matching (AGM)**
  - Downsizing the search area by $^1/_3$ according to the threshold of the geomagnetic profile index (σ_Z)









Simulation Case 2

Numerical simulations

- **Summary of numerical simulation results**

Summary of the simulation results

Simulation Case	Measurements	Longitudinal direction	Latitudinal direction
1	Matching rate (%)	99.6	
	Mean error (m)	2.46	-2.46
	Standard deviation (m)	95.63	95.63
	DRMS (m)	135	
2 : without AGM	Matching rate (%)	97.6	
	Mean error(m)	130.8	-385.2
	Standard deviation (m)	702.1	819.9
	DRMS (m)	1079	
2 : with AGM	***Matching rate (%)***	***99.3***	
	Mean error(m)	***19.7***	***-44.4***
	Standard deviation (m)	***117.1***	***702.7***
	DRMS (m)	***712.4***	

Concluding remarks

- **Conclusions**
  - Proposing a new approach to GAN based on two symmetric CNNs combined with NCC
    - ***Applicable to new area where geomagnetic information can be provided***
      - By removing dependency between NNs complexity and navigation area
      - Possible to estimate position using only measurement pattern and small search maps
    - ***Maintaining matching performance under unexpected conditions***
      - By combining abstraction and refinement steps for gradual matching
        - Over 98% matching rate even if the size of the search area is more than twice
        - Over 96% matching rate even if the measurement noise is 4-times higher than training noise
    - ***Improving position estimation error under less changing profile***
      - By adjusting the size of the search area according to σ_Z threshold
        - Reducing the position estimation error 35% even if the geomagnetic profile is less change

Concluding remarks

- **Contributions**
  - ***Proposing a two symmetric CNNs architecture to geomagnetic matching for navigation is an innovative idea***
  - ***Expanding navigation area to the globe in NN-based GAN***
    - Removing dependency between NNs complexity and navigation area
    - Possible to matching with the measurements from the untrained area
    - Maintaining high matching performance with the measurements corrupted by the unexpected noise
  - ***Improving position estimation error even if under the less changing geomagnetic profile***
    - Introduced the index to quantify geomagnetic profile of the search area
    - Improving position estimation error using proposed AGM algorithm

ASCL
Aerospace Systems & Control Lab.

References

[1] Zhou, J., Liu, Y., Ge, Z.: ‘Geomagnetic Matching Algorithm Based on Probabilistic Neural Network’. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2010, 225, (1), pp. 120–126.

[2] Anderson, E.W.: ‘The Principles of Navigation’, Hollis and Carter, 1966.

[3] Tyrén, C.: ‘Magnetic Terrain Navigation’. Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology, Durham, NH, USA, June 1987, pp. 245–256.

[4] Lee, J., Kwon, J., Yu, M.: ‘Performance Evaluation and Requirements Assessment for Gravity Gradient Referenced Navigation’, MDPI sensor, 13 July 2015, 15, (7), doi: 10.3390/s150716833.

[5] Xie, W., Qu, Z., Li, Q.: ‘A Fast Algorithm of the Geomagnetic Correlation Matching based on MSD’. in Proc. 3rd Int. Conf. Control, Automat. Syst. Eng. (CASE), Atlantis Press, 2013, 45, pp. 59–62.

[6] Zhang, Y.: ‘Geomagnetism-aided Navigation based on Matching Algorithm for Underwater Vehicles.', In Selected and Revised Results of the 2011 International Conference on Mechanical Engineering and Technology, November 2012, pp. 59–65.

[7] Chen, Z., Zhang, Q., Pan, M., Chen, D., Wan, C., Wu, F., Liu, Y.: ‘A New Geomagnetic Matching Navigation Method based on Multidimensional Vector Elements of Earth’s Magnetic Field’, IEEE Geosci. Remote Sens. Lett. 2018, 15, (8), pp. 1289–1293.

[8] Feng, J., Chen, G., Tianyou, L.: ‘Kalman Filter Underwater Passive Geomagnetic Navigation Technology Research Considering the Impact of Diurnal Variation’, Image and Signal Processing 2nd International Congress on, October 2009, pp. 1–4.

[9] Mu, H., Wu, M., Hu, X., Ma, H.: ‘Geomagnetic Surface Navigation using Adaptive EKF’. Proceedings of the 2nd IEEE Conference on Industrial Electronics and Applications, Harbin, China, May 2007, pp. 2821–2825.

[10] Canciani, A., C., Raquet, J.: ‘Airborne Magnetic Anomaly Navigation’, IEEE Trans. Aerosp. Electron. Syst., 2017, 53, (1), pp. 67–80.

[11] Bromley, J., Guyon, I., LeCun, Y., Sackinger, E., Shah. R.: ‘Signature Verification using a Siamese Time Delay Neural Network’. Proceedings of the 6th International Conference on Neural Information Processing Systems, Denver, CO, USA, December 1993, pp. 737–744.

[12] Bergman, N.: ‘Recursive Bayesian Estimation: Navigation and Tracking Applications’. PhD dissertation, Linköping University, 1999.

[13] Yamashita, R., Nishio, M., Do, R., Togashi, K.: ‘Convolutional Neural Networks: An Overview and Application in Radiology’, Insights Imaging, 2018, 9, (611), https://doi.org/10.1007/s13244-018-0639-9.

APPENDIX

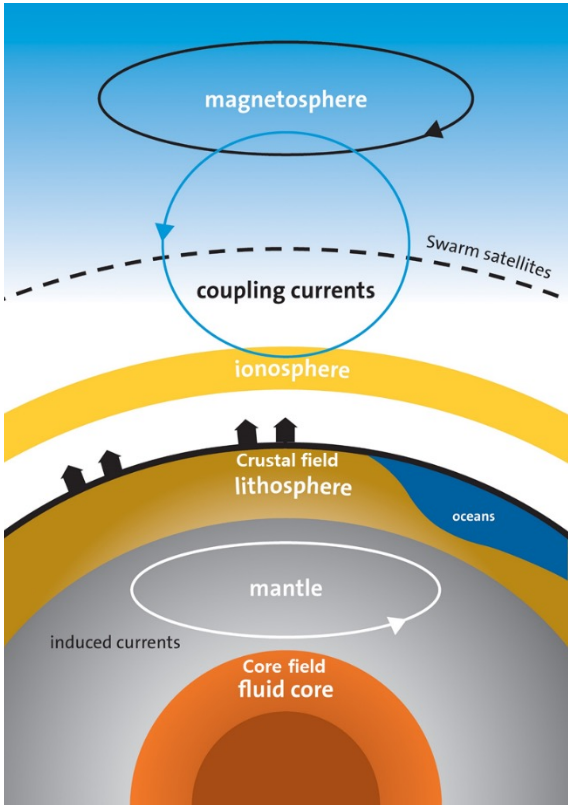
Geomagnetic Matching
for Navigation based on Neural Networks

Appendix

- **Definition of geomagnetism**
  - The sum of the Earth magnetic fields

Earth Magnetic Field

- **Main Field: Internal field**
  - **Core field**
    - *Rotating of conducting fluid outer core*
    - 95~99% (25,000–65,000 nT(1))
  - **Crustal field**
    - Earth's crust and upper mantle
    - Generated by mineral components
    - 1~5% (Variation depends on mineral)
  - **Induced current**
    - Coupling main field and external field
- **Disturbance Field: External field**
  - **Ionosphere field**
    - Solar wind effect (Normal variation: 20-40 nT)
    - *Equatorial electrojet: 100–200 nT/24hr*
    - *Auroral current: 1500 nT/27–day at pole*
    - Solar quiet variation: 24 hr period
  - **Magnetosphere field**
    - Ring current by charged particle from sun
    - 20–30 nT/quiet day
    - Hundreds of nT when coronal mass ejection
  - **Coupling current**



Earth magnetic fields [1]

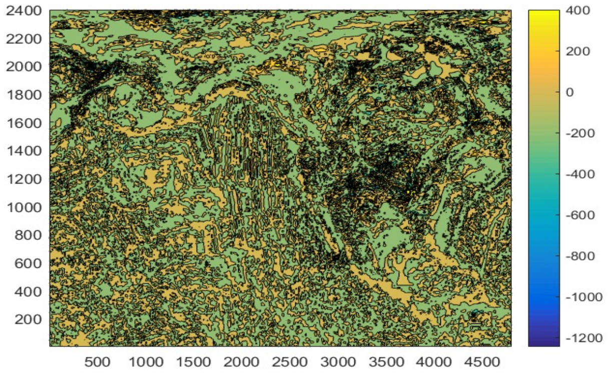
(1)nT: nano tesla ($1\ \mathrm{G} = 10^{-4}\ \mathrm{T}$)

[1] http://www.goodrichscience.com/2-gc-magnetic-force-field.html

Appendix

- **Geomagnetic reference map**
  - Geomagnetic anomaly
    - ***Geomagnetic Intensity deviated by crustal field***
      - Knowledge of subsurface structure and composition of the Earth's crust
      - ***Used for resource exploration and navigational information***
    - Geomagnetic anomaly is used as geomagnetic reference map
      - Ex) World digital magnetic anomaly map (WDMAM), Earth magnetic anomaly grid (EMAG)
  - Earth magnetic anomaly grid 2 (EMAG 2)

Category	Quantity	Unit
Grid resolution	2	arcmin
	~3.7	km
Total number of grids	4800 × 2400	grid
Total map size	17,779 × 8,889	km^2



EMAG 2

Appendix

- **PNN**
  - ***Classification-based*** approach to GAN [1]
    - NN for Gaussian mixture model
      - ***Pattern layer**: model of conditional pdf*
      - ***Summation layer**:* sums the outputs of the pattern layer
      - ***Output layer**:* select a maximum probability for classification

$$h_{x_i} = \frac{1}{n}\sum_{j=1}^{n} H_{x_i,j}$$

$$H_{x_i,j} = \exp\left(-\frac{\left\| \mathbf{M} - \mathbf{M}_{x_i,j} \right\|_2^2}{\left(\sigma_{x_i,j}\right)^2} \right)$$

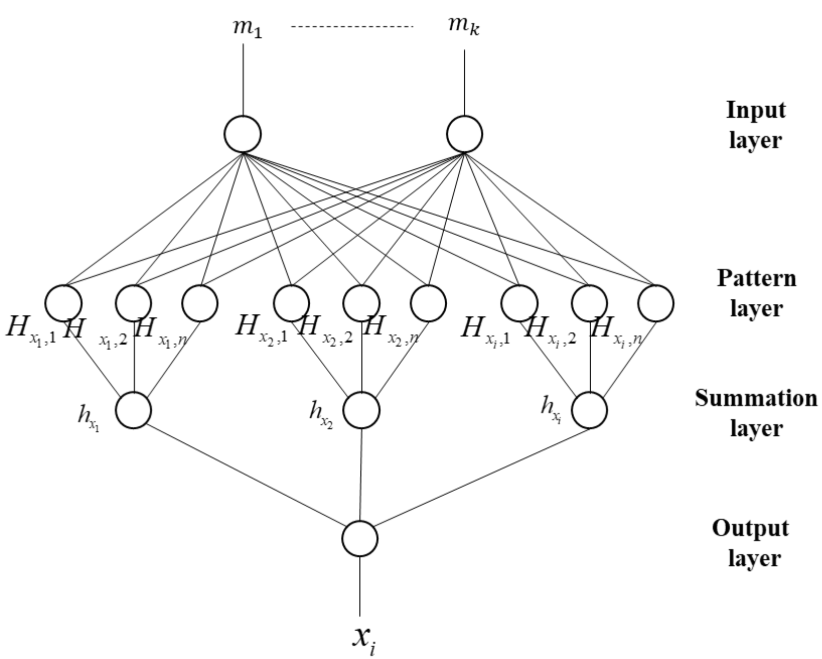
$$x_i = \arg\max_i h_{x_i}$$

x_i : i-th position class

$\mathbf{M}$: Input geomagnetic measurment vector

$\mathbf{M}_{x_i,j}$: j-th mean parameter vector for x_i class

$\sigma_{x_i,j}$: j-th variance parameter for x_i class



[1] Zhou, J., Liu, Y., Ge, Z.: 'Geomagnetic matching algorithm based on probabilistic neural network'. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2010, 225, (1), pp. 120–126.

Appendix

- **PNN**
  - Characteristics of classification-based approach
    - ***Increase NN complexity***
      - As the position class increases, the complexity of the NN increases
    - ***Impossible to classify measurements coming from a new area***
      - Applicable to only pre-trained area
    - ***A small number of measurement samples at a classified position***
      - Required many samples to train NNs

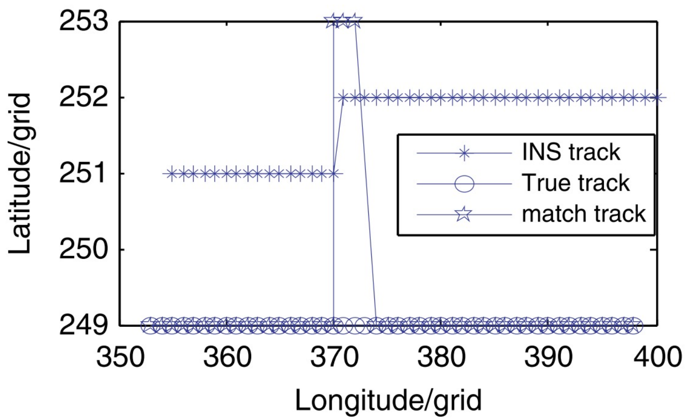


Fig. 5 Matching results for the smooth simulation area I

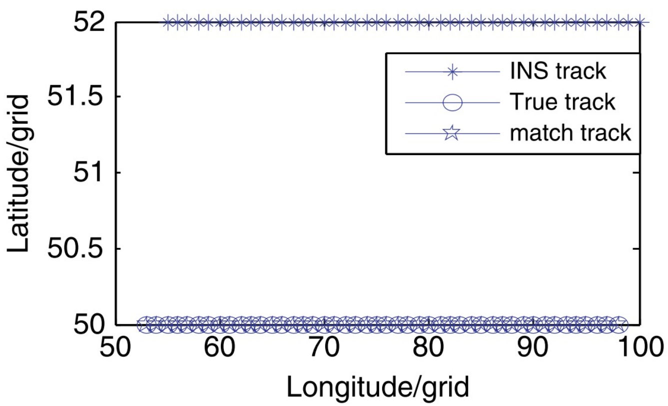


Fig. 6 Matching results for the rough simulation area II

Appendix

- **CNNs**
  - ***Abstraction: dimension reduction***
    - Transform from ***complex data of input space to simple feature data of target space***

CNNs

$$G_W(\cdot): \mathbb{R}^D \rightarrow \mathbb{R}^d, \qquad D \gg d$$

- Adaptively ***learn spatial hierarchies of features by training*** [13]
  - Recursively update learnable weight vector ($\mathbf{W}$) to minimize the loss function with datasets

Convolution

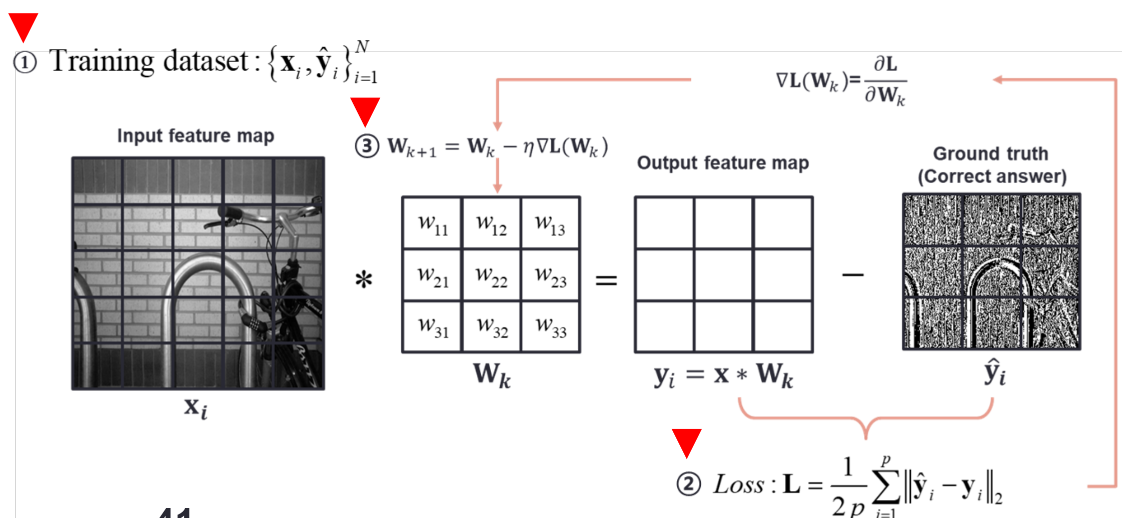
$$G_W(\mathbf{x}): \ \mathbf{y} = \mathbf{x} * \mathbf{W}$$

$$\mathbf{y}(i,j) = \sum_{u=0}^{l-1}\sum_{v=0}^{l-1} \mathbf{x}(i+u, j+v)\mathbf{W}(u,v)$$

$\mathbf{x}$: Input feature map

$\mathbf{W}$: Kernel or filter (learnable weight vector)

$\mathbf{y}$: Output feature map



Appendix

- **CNNs**
  - Automatically and adaptively ***learn spatial hierarchies of features through training*** [13]

Convolutional operation

$$G_W(\mathbf{x}):\ \mathbf{y} = \mathbf{x} * \mathbf{W}$$

$$\mathbf{y}(i,j) = \sum_{u=0}^{l-1} \sum_{v=0}^{l-1} \mathbf{x}(i+u, j+v)\mathbf{W}(u,v)$$

$\mathbf{x}$: Input feature map

$\mathbf{W}$: Kernel or filter (learnable weight vector)

$\mathbf{y}$: Output feature map

Input feature map $\mathbf{x}$

$*$

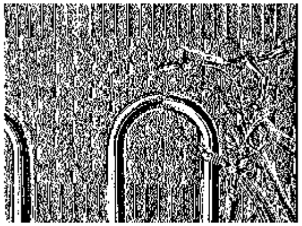
Kernel $\mathbf{W}$

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

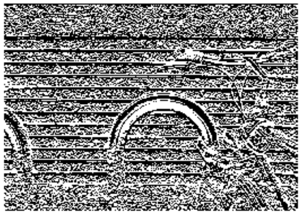
$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

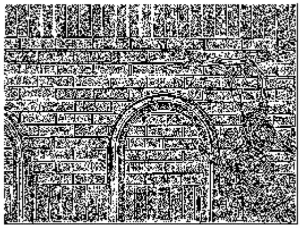
Output feature map $\mathbf{y}$



Vertical feature



Horizontal feature



Edge feature

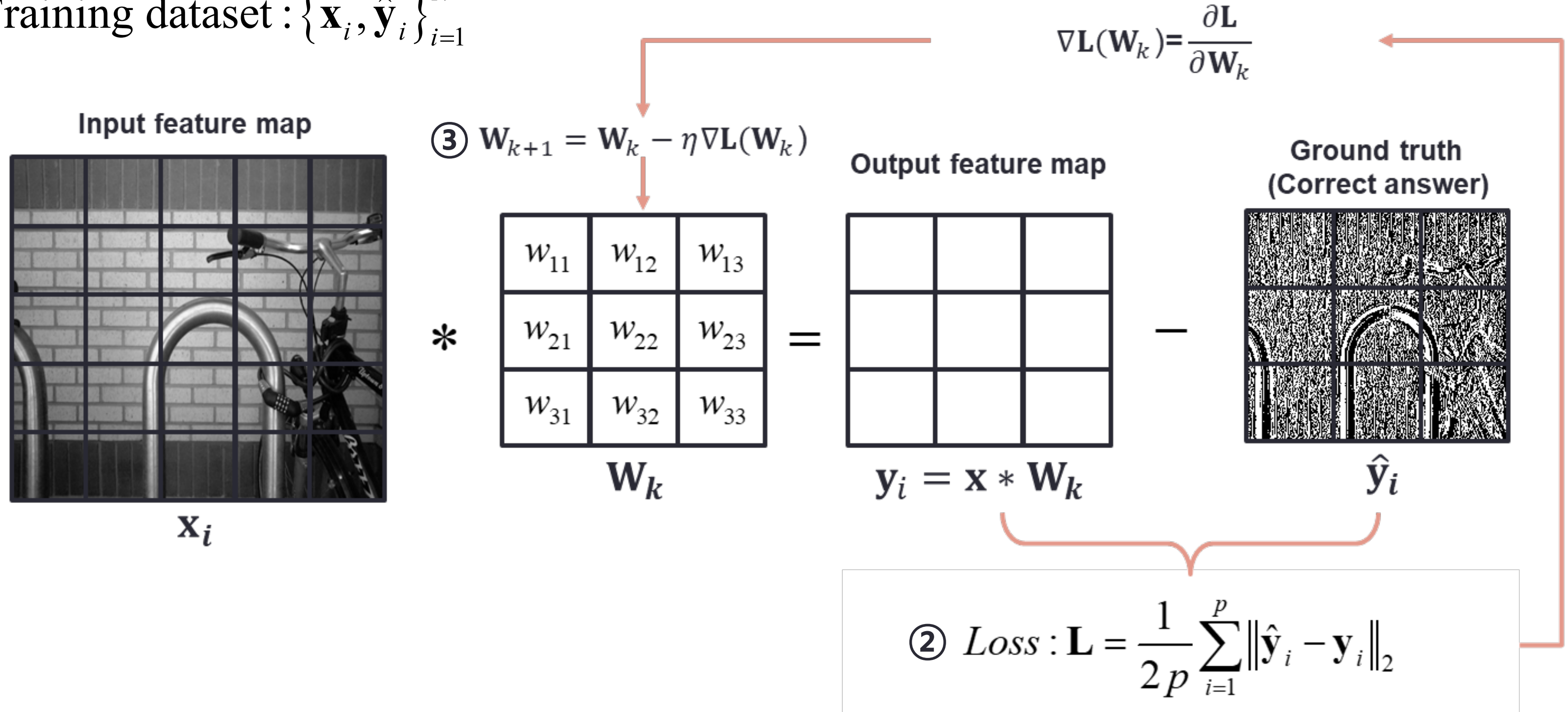
[14] Yamashita, R., Nishio, M., Togashi, K.: 'Convolutional neural networks: an overview and application in radiology', doi:https://doi.org/10.1007/s13244-018-0639-9.

$\mathbf{x}_i$: Input feature maps
$\hat{\mathbf{y}}_i$: Ground truth
$\mathbf{y}_i$: Output feature map
$\mathbf{W}_k$: Kernel (learnable weight vector)
η: Learning rate ($0 < \eta \leq 1$)
i: Sample index, $i = 1, \ldots, N$
p: Batch size
$\mathbf{L}$: Loss function

Appendix

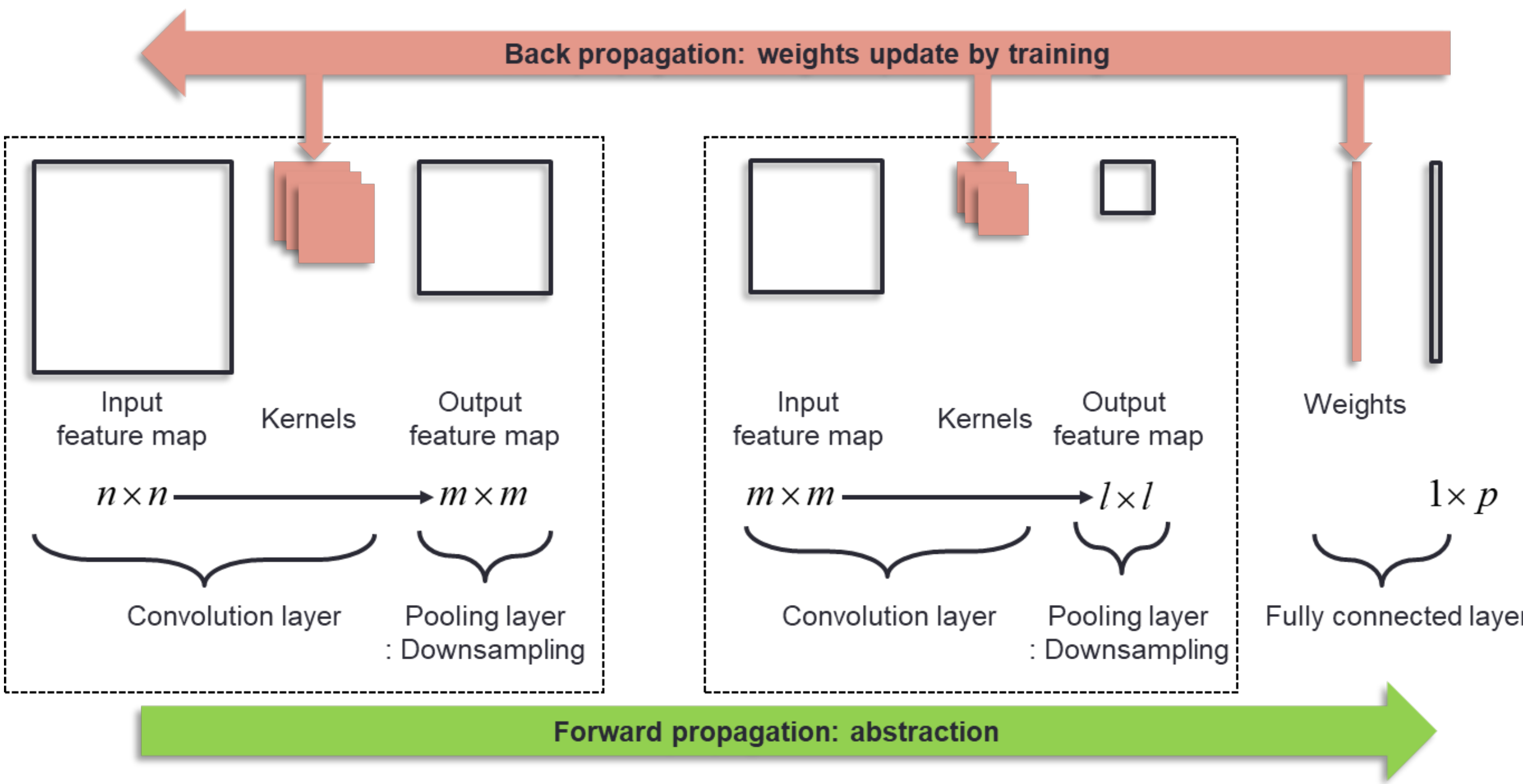
- **CNNs**
  - Training CNNs
    - ***Recursively update weight vector (W) to minimize the loss function***
      - Required ***datasets, loss function***, and ***optimization methods***

① Training dataset : $\{\mathbf{x}_i, \hat{\mathbf{y}}_i\}_{i=1}^{N}$



Appendix

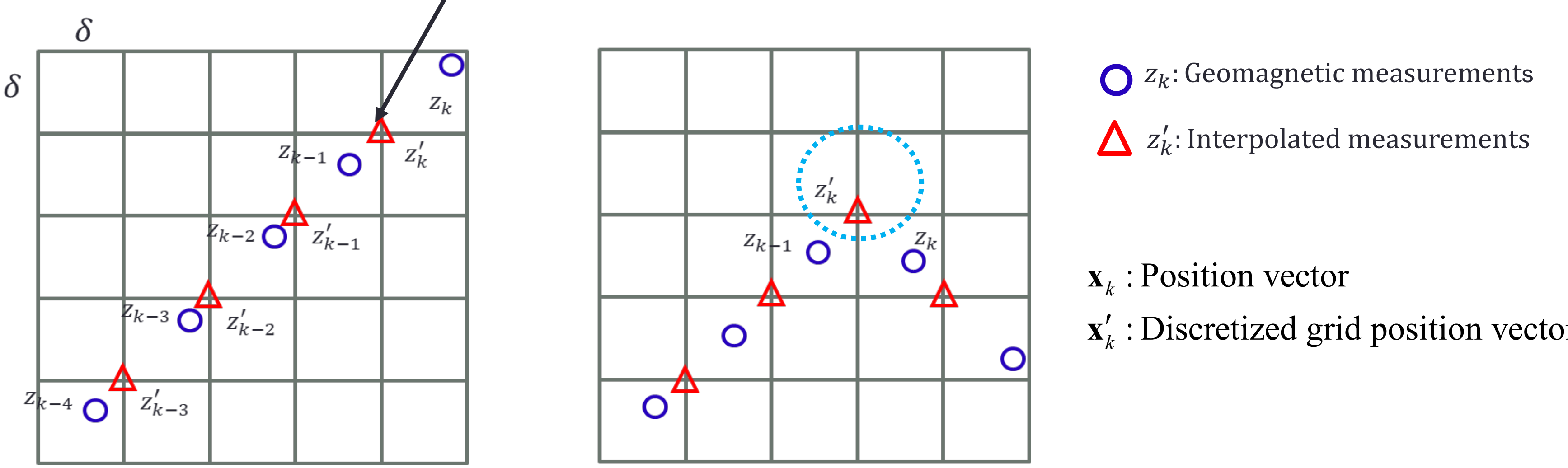
- **CNNs**
  - ***Stacking of convolution, pooling, and fully connected layers***
    - Pooling layer: reduce the spatial dimensions by down sampling
    - Fully connected layer: maps the extracted features into final output
  - Dimension reduction: $G_W(\mathbf{x}): \mathbb{R}^{n\times n} \to \mathbb{R}^{1\times p}$



Appendix

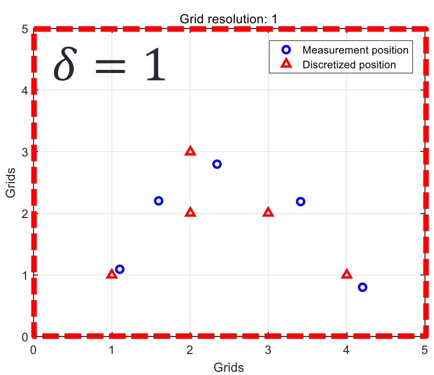
- **Grid adjustment**
  - Assigning geomagnetic measurements to discrete grid points with the grid resolution (δ)
    - The geomagnetic measurements are measured on arbitrary positions
    - Discrete input size of CNNs: $n \times n$ grids with the grid resolution (δ)

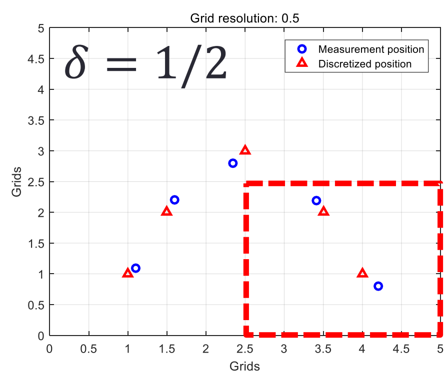
$$z'_k = f\left(\mathbf{x}'_k\right) = z_{k-1} + \frac{\left(\mathbf{x}'_k - \mathbf{x}_{k-1}\right)}{\left(\mathbf{x}_k - \mathbf{x}_{k-1}\right)}\left(z_k - z_{k-1}\right)$$

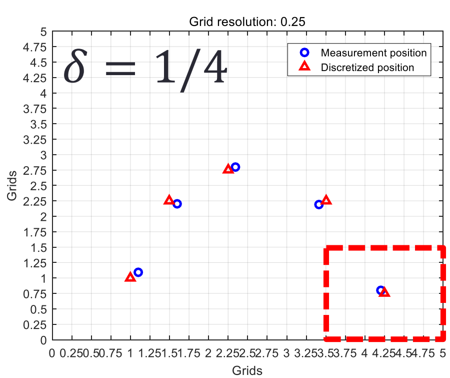


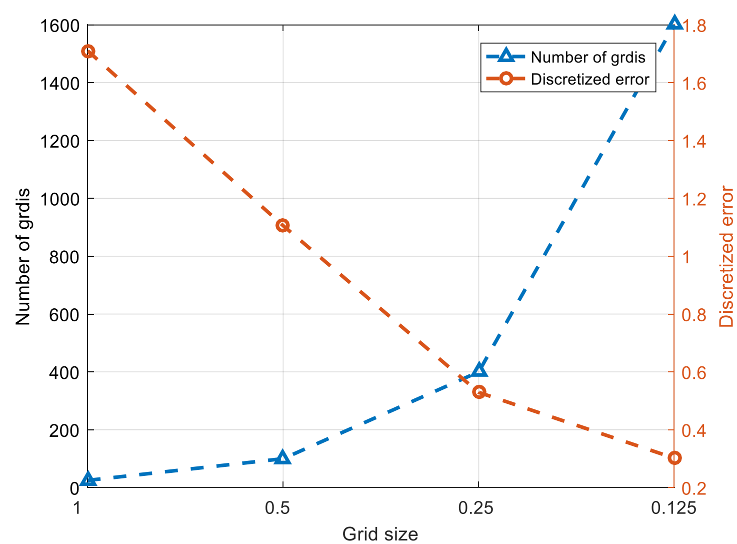
Appendix

- **Grid adjustment**
  - Calculate minimum grid resolution $\delta = g_k$
  - Calculate the number of measurements to be included $\mathbf{M}_k$
  - Assign the measurements to discrete grid points
    - smaller grid resolution, lower discretization error ➔ fewer samples









Appendix

- **Grid adjustment**

Calculate the grid resolution
$g_k = \Delta x_{\min} \times SF$

↓

Calculate the number of measurements
$N : l_{\max} \le n \times g_k$

↓

Assign the measurements to discrete grid positions
$\mathbf{M}_k\left(\mathbf{x}'_{k-j}\right) \leftarrow m_k(j)$

<u>Calculate the grid resolution</u>

$$\Delta \mathbf{x}_j = \mathbf{x}_{k-j} - \mathbf{x}_{(k-1)-j},\ j = 0,\ldots,N$$

$$\Delta \mathbf{X}_k = \left\{\Delta \mathbf{x}_{k-j}\right\}_{j=0}^{N}$$

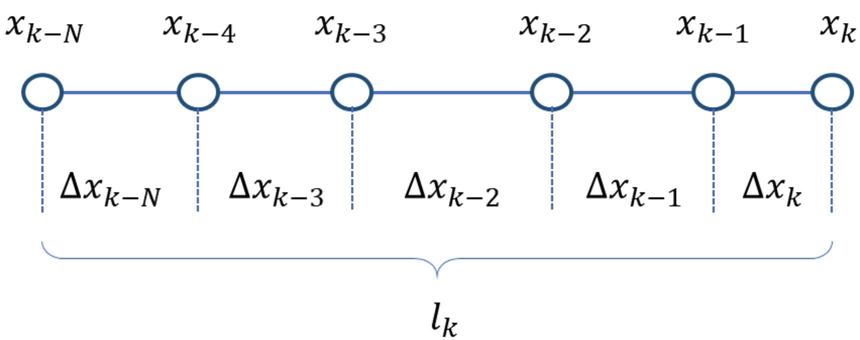
$$\Delta x_{\min} = \min_{\mathbf{x}_{th} \le \Delta \mathbf{X}_k} \Delta \mathbf{X}_k$$

$$g_k = \Delta x_{\min} \times SF$$

$\mathbf{x}_k$: Position vector corresponding to z_k
$\Delta \mathbf{x}_k$: Differentiated position vector
$\Delta \mathbf{X}_k$: Set of $\Delta \mathbf{x}_k$
N : number of measurements
$\Delta x_{\min}$: Minimum differentiated position value
$\mathbf{x}_{th}$: Threshold for $\Delta x_{\min}$
g_k: Grid resolution
SF : Scale factor, $0 < SF \le 1$

Appendix

- **Grid adjustment**



$$l_k \leq \text{Size of CNNs}$$

$$N = \begin{cases} N-1, & \text{if } \mathrm{l_k} > \text{size of CNNs} \\ N+1, & \text{if } \mathrm{l_k} \leq \text{size of CNNs} - 1 \end{cases}$$

<u>Calculate the number of measurements</u>

$$\mathbf{l}_k = \sum_{j=0}^{N} \Delta \mathbf{x}_{k-j}$$

$$l_{\max} = \max_i \mathbf{l}_k(i)$$

$$N : l_{\max} \leq n \times g_k$$

$$\mathbf{m}_k = \left\{ z_{k-j} \right\}_{j=0}^{N}$$

$$\mathbf{X}_k = \left\{ \mathbf{x}_{k-j} \right\}_{j=0}^{N}$$

$\mathbf{l}_k$: Total length vector

$\mathbf{m}_k$: Set of measurements

$\mathbf{X}_\mathrm{k}$:Set of position vectors corresponding to z_k

Appendix

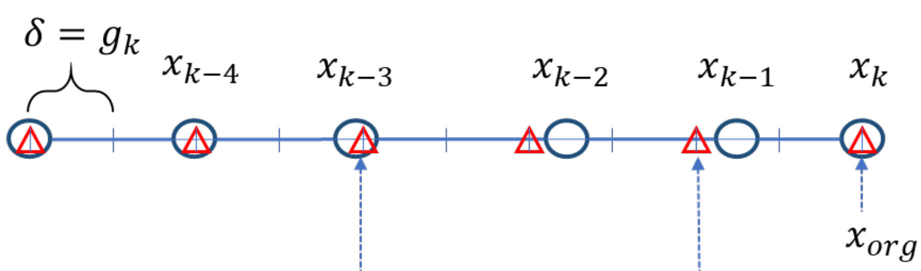
- **Grid adjustment**

Assign the measurements to discrete grid position

$$\mathbf{x}_{org} = \left[x_m \;\; y_m\right]^T \qquad \begin{cases} x_m = \min\limits_j \left(x_{k-j}(0)\right) \\ y_m = \min\limits_j \left(x_{k-j}(1)\right) \end{cases}$$

$$\mathbf{x}'_{k-j} = \begin{cases} \left\lfloor \left(\mathbf{x}_{k-j} - \mathbf{x}_{org}\right)/g_k \right\rfloor, & \text{if } \left(\left(\mathbf{x}_{k-j} - \mathbf{x}_{org}\right)\% g_k\right) \le 0.5 \\ \left\lceil \left(\mathbf{x}_{k-j} - \mathbf{x}_{org}\right)/g_k \right\rceil, & \text{otherwise} \end{cases}$$

$$\mathbf{M}_k\left(\mathbf{x}'_{k-j}\right) \leftarrow m_k(j) \qquad j = [0, \ldots, N]$$



$\mathbf{x}_{org}$: origin position vector

$\mathbf{x}'_k$: discretized grid position of $\mathbf{x}_k$

$$x'_{k-j} = \left\lceil \left(\mathbf{x}_{k-1} - \mathbf{x}_{org}\right)/g_k \right\rceil, \text{ if } \left(\left(\mathbf{x}_{k-j} - \mathbf{x}_{org}\right)\% g_k\right) > 0.5$$

$$\mathbf{x}'_{k-3} = \left\lfloor \left(\mathbf{x}_{k-3} - \mathbf{x}_{org}\right)/g_k \right\rfloor, \text{ if } \left(\left(\mathbf{x}_{k-j} - \mathbf{x}_{org}\right)\% g_k\right) \le 0.5$$

Appendix

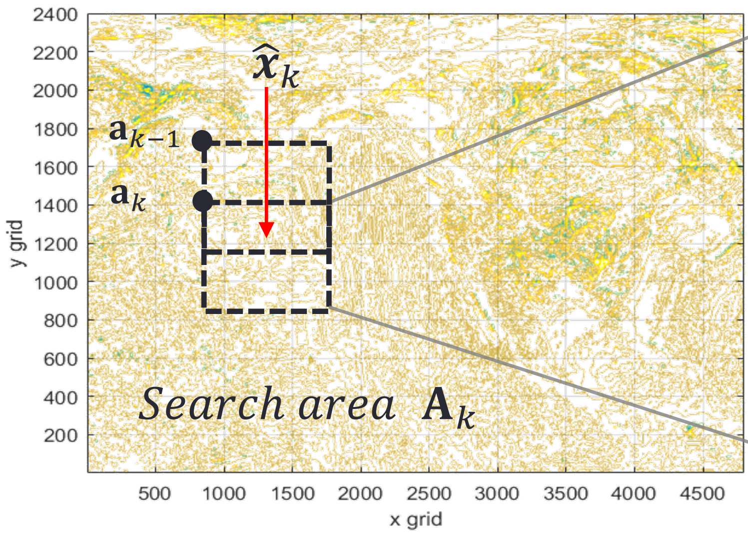
- **Small search maps (S_k^i)**
  - Search area ($\mathbf{A}_k$)
    - $\mathbf{A}_k$ is defined at $\mathbf{a}_k$having $m \times m$ size with δ where $m > n$
      - The origin position ($\mathbf{a}_k$) updated at every discrete time k

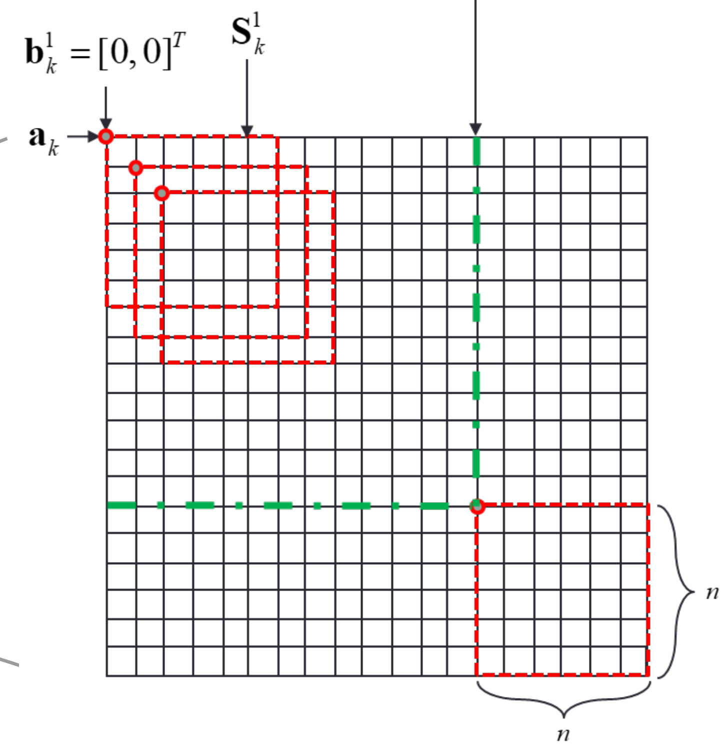
$$\boxed{\mathbf{a}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{u}_k \cdot dT - \frac{g_k}{2}\left[m, m\right]^T}$$

$\hat{\mathbf{x}}_k$: Estimated postion vector

$\mathbf{u}_k$: Velocity vector

$$[0,0]^T \le \mathbf{b}_k^i \le \left[(m-n)-1, (m-n)-1\right]^T$$





Appendix

- **Training two symmetric CNNs**
  - Training datasets

Training datasets

Matched data set: $\mathbf{D}_M = \left\{ \left(Y, \mathbf{M}, \mathbf{S} \right)^i \right\}_{i=1}^{N}$

Mismatched data set: $\mathbf{D}_S = \left\{ \left(Y, \mathbf{M}, \mathbf{S}' \right)^i \right\}_{i=1}^{N}$

$$Y = \begin{cases} 0, & \text{if matched data pair} \\ 1, & \text{if mismatched data pair} \end{cases}$$

Y : binary label

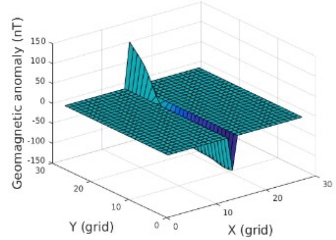
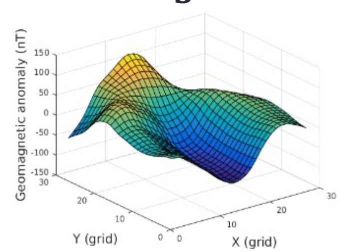
$(Y, \mathbf{M}, \mathbf{S})$: Matched data pair

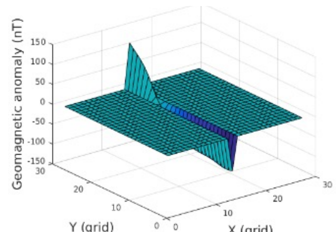
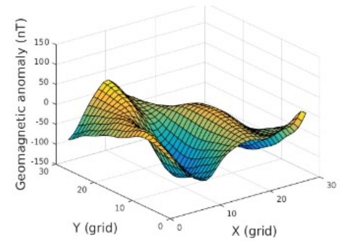
$(Y, \mathbf{M}, \mathbf{S}')$: Mismatched data pair

$\mathbf{M}$: Mesurement pattern

$\mathbf{S}$: Search map maeched to $\mathbf{M}$

$\mathbf{S}'$: Search map mismaeched to $\mathbf{M}$

	Binary label	$\mathbf{M}$	$\mathbf{S}$
Matched data set	$Y = 0$		

	Binary label	$\mathbf{M}$	$\mathbf{S}'$
Mismatched data set	$Y = 1$		

Appendix

- **Training datasets**

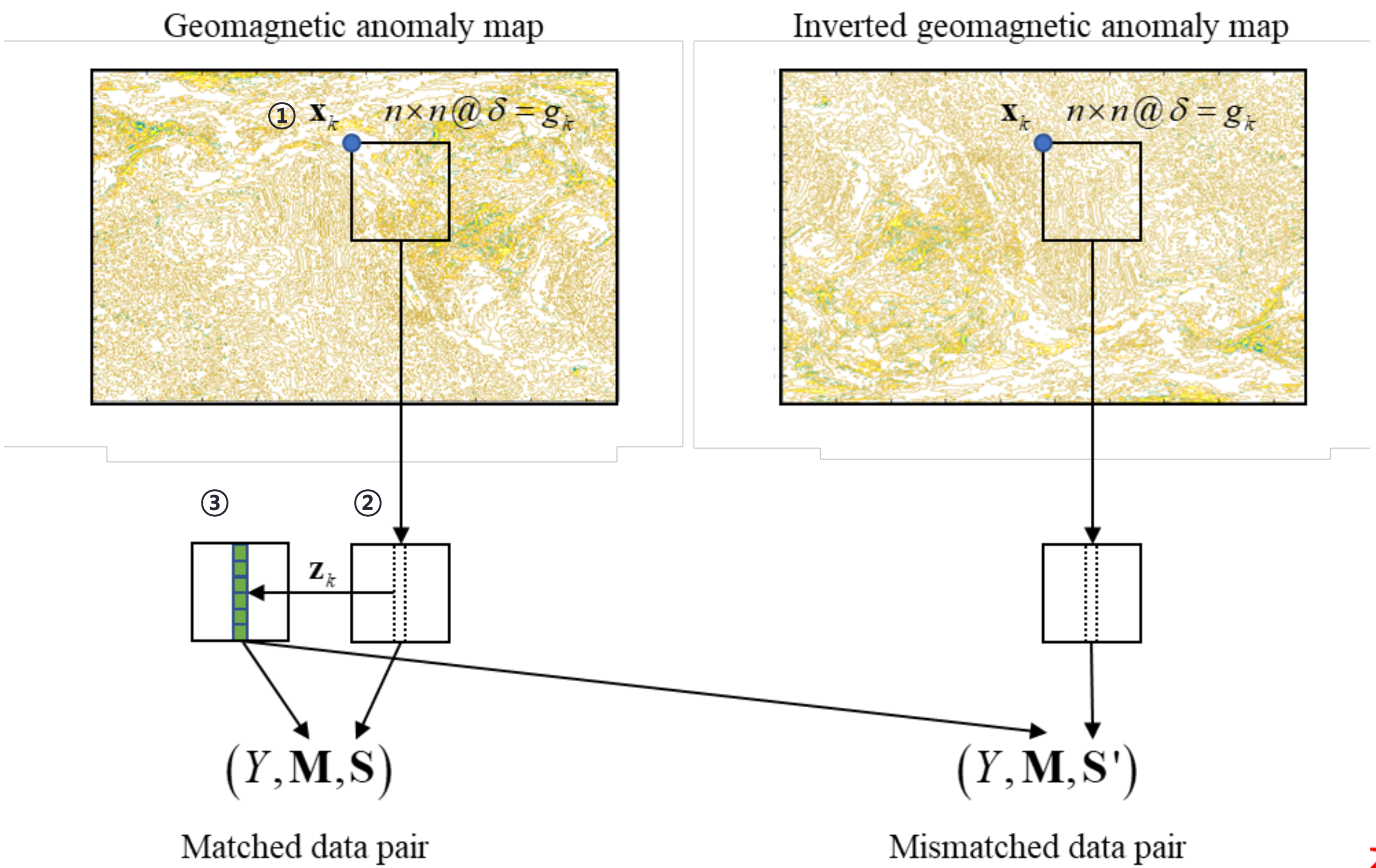
$\mathbf{x}_k$: position vector

$\mathbf{u}_k$: velocity

dT : sampling period

$\mathbf{w}_k$: zeromean Gaussian noise

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{u}_k \cdot dT + \mathbf{w}_k$$

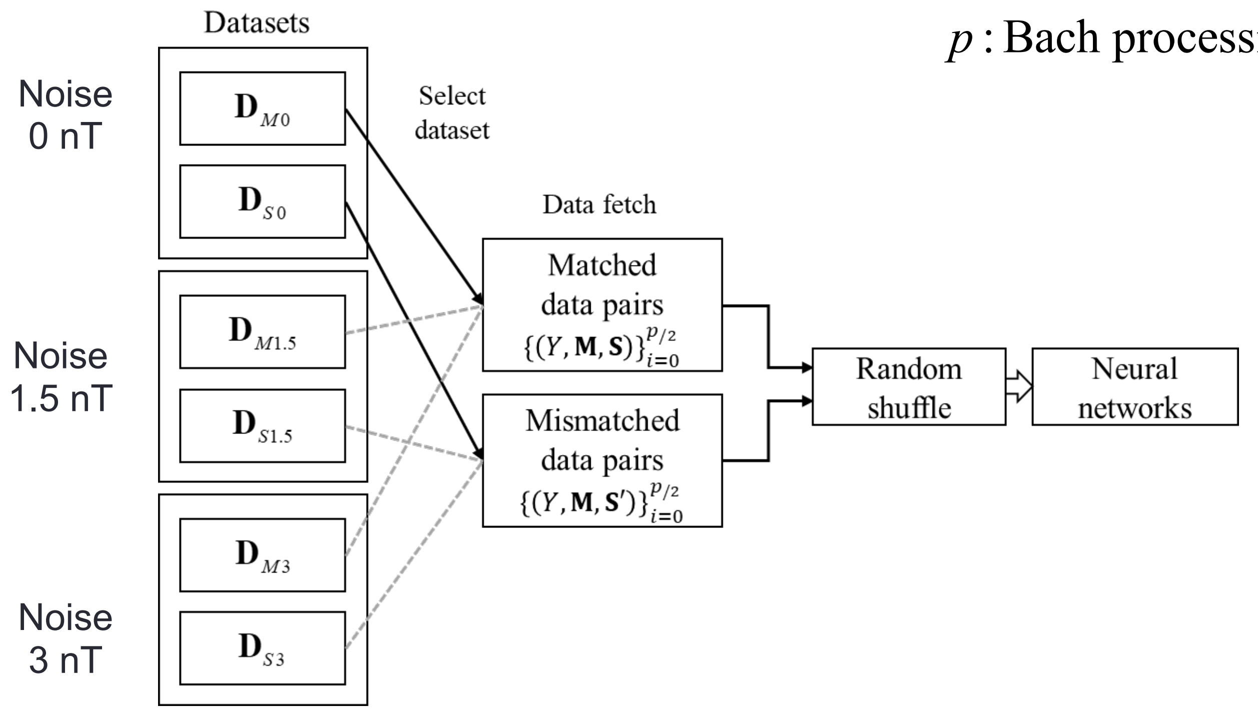


Appendix

- **Training scheme**

$$L(\mathbf{W}) = \sum_{i=1}^{p} l\left(\mathbf{W}, (Y, \mathbf{M}, \mathbf{S})^{i}\right)$$

p : Bach processing size



Appendix

- **Specification of two symmetric CNNs**
  - ***5 convolutional layers + 1 fully connected layer***

Layer	Kernel size	Feature map	Stride	Output size
C1	7×7	32	1	28×28
P2	2×2	32	1	14×14
C3	5×5	64	1	14×14
P4	2×2	64	1	7×7
C5	3×3	128	1	7×7
P6	2×2	128	1	4×4
C7	7×7	256	1	4×4
P8	2×2	256	1	2×2
C9	1×1	24	1	2×2
P10	2×2	24	1	1×1
F11	-	-	-	1×24

Appendix

- **Validation check for training datasets**
  - Validation of generated training datasets
    - Why: large amounts of data are generated by the program in batch
    - $\mathbf{D}_{M3}$: almost zero mean and small deviation
    - $\mathbf{D}_{S3}$: biased but wide deviation

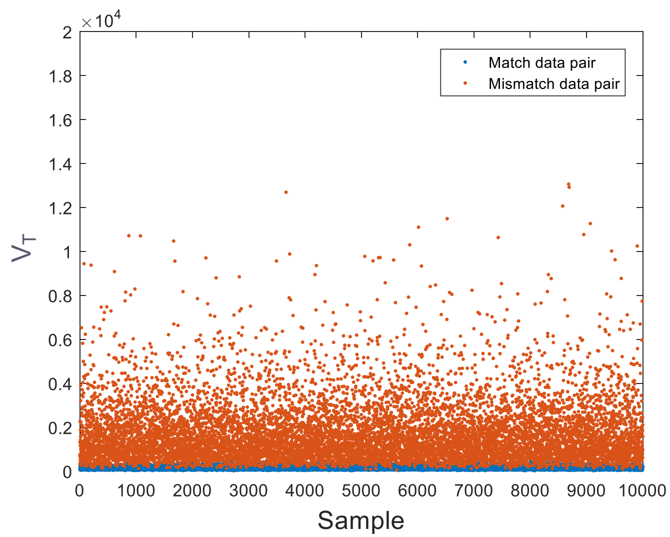
Validation training datasets

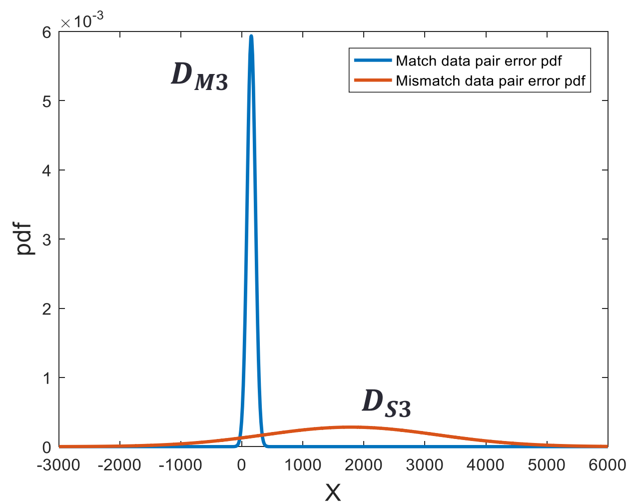
$$\mathbf{V}_T = |\mathbf{M}| - \sqrt{\mathbf{M} \odot \mathbf{S}}$$

$\mathbf{M}$: measurement map

$\mathbf{S}$: Search map

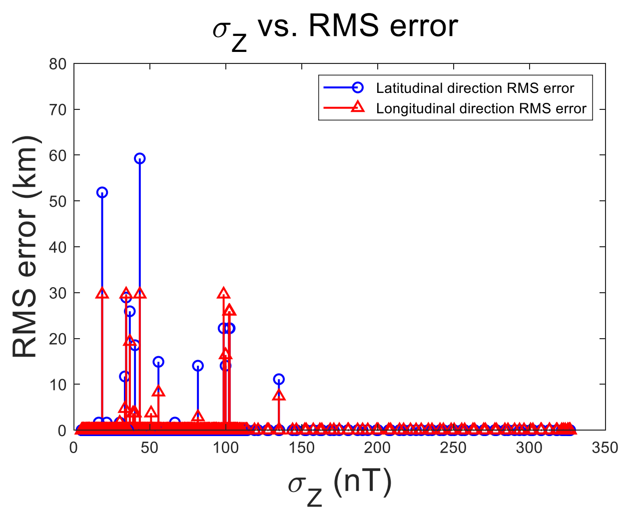
$\odot$: component wise multiplication

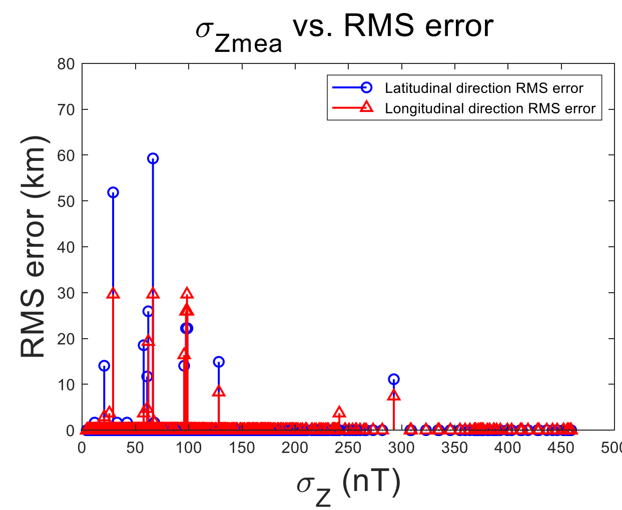




Appendix

- **Geomagnetic profile index**
  - Presenting σ_Z threshold using 5 simulation cases





σ_Z index
with search area at every k

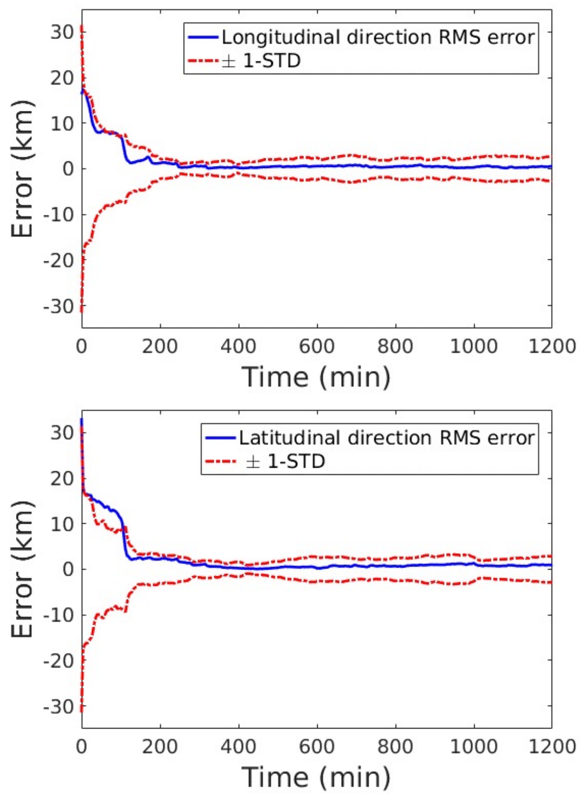
σ_Z index
with geomagnetic measurement map at every k

: Much clear threshold for adaptive geomagnetic matching

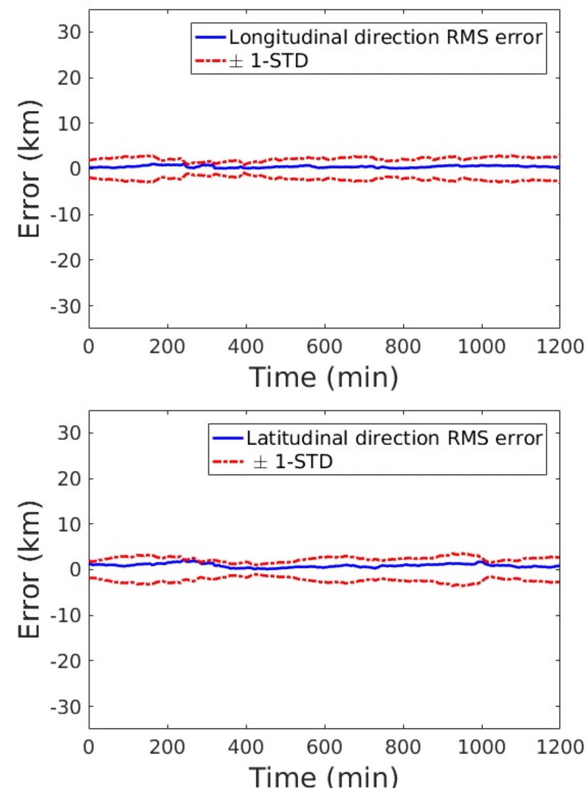
: Outliers are occurred due to lack of measurements

Appendix

- **Search mode operation**
  - Provide initial position for PF in wide search area at the beginning stage of estimation



(a) SISR[(1)] only



(b) SISR initialized by the proposed algorithm

(1) Sequential importance sampling with resampling

Appendix

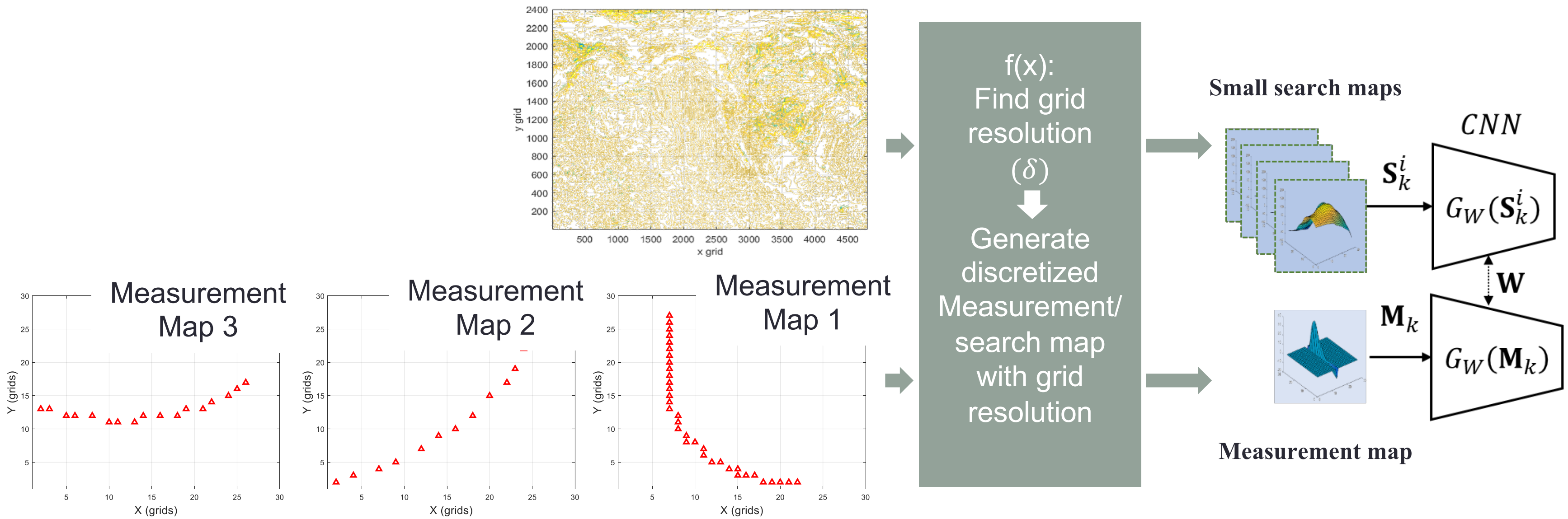
- **Search mode operation**

Simulation results of particle filter application

Simulation case	Measurements	Longitudinal direction	Latitudinal direction
SISR only	Convergence time (min)	295	
	Mean (m)	-313	-1163
	Standard deviation (m)	2318	2093
	DRMS (m)	3123	
SISR initialized by the proposed algorithm	Convergence time (min)	0 (112)	
	Mean (m)	-46	-482
	Standard deviation (m)	229	236
	DRMS (m)	329	

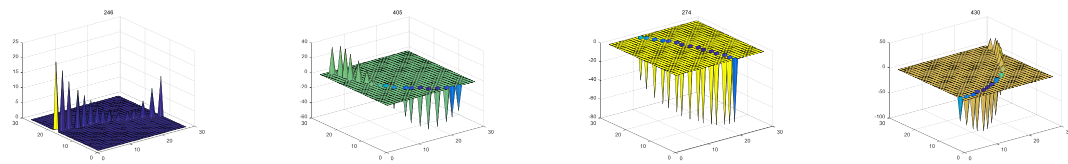
Appendix

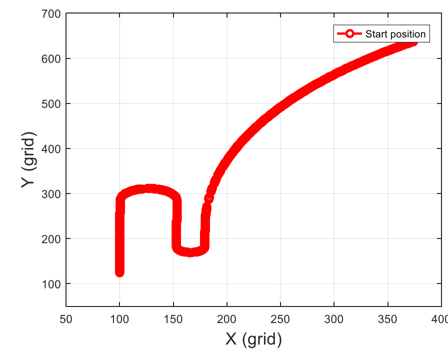
- **Geomagnetic navigation under dynamic pattern**
  - ***Generating the measurement pattern***
    - Generating measurement pattern with the size of 28 × 28 grids using discrete grid resolution according to vessels' movement
  - ***Training two symmetric CNNs***
    - Generating datasets according to simulated vessels' movement
    - Training whole patterns to satisfied target loss



Appendix

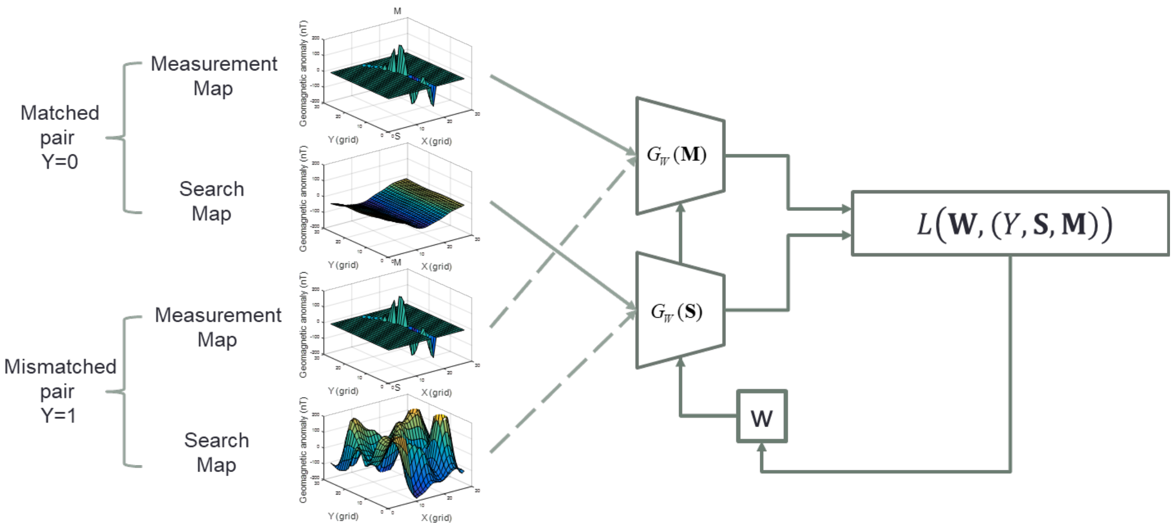
- **Training two symmetric CNNs**
  - Generate categorized datasets under controlled dynamic condition
    - Four representative patterns are generated





  - Training two symmetric CNNs
    - 1st :Train categorized pattern independently
    - 2nd : Transfer learning to add up whole patterns

- **Sampling period: 1 min**
- **Total sampling point: 1,000**
- **Vessel velocity: 30knot**



Loss function:

$$L(\mathbf{W},(Y,\mathbf{S},\mathbf{M})) = (1-Y)\times L_M(D_W(\mathbf{S},\mathbf{M})) + Y \times L_S(D_W(\mathbf{S},\mathbf{M}))$$

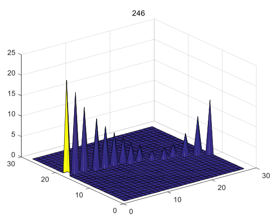
$$D_W(\mathbf{S},\mathbf{M}) = \|G_W(\mathbf{S}) - G_W(\mathbf{M})\|$$

$$L_M(D_W(\mathbf{S},\mathbf{M})) = \max\left((m - D_W(\mathbf{S},\mathbf{M})), 0\right)$$

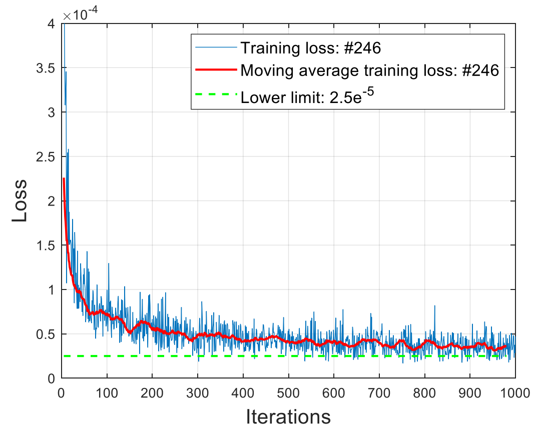
$$L_S(D_W(\mathbf{S},\mathbf{M})) = \|G_W(\mathbf{S}) - G_W(\mathbf{M})\|$$

Appendix

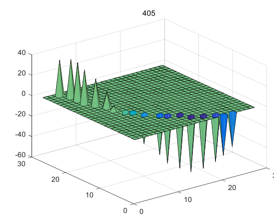
- **Training two symmetric CNNs**
  - Training two symmetric CNNs independently with categorized pattern



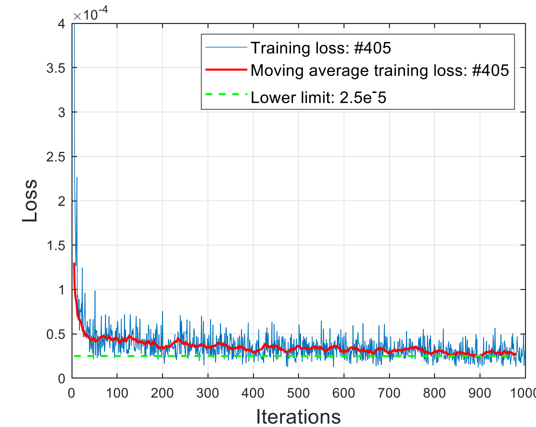
#246



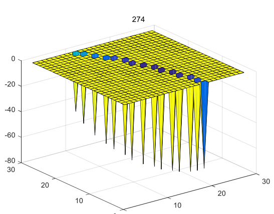
Validation error 7.61%



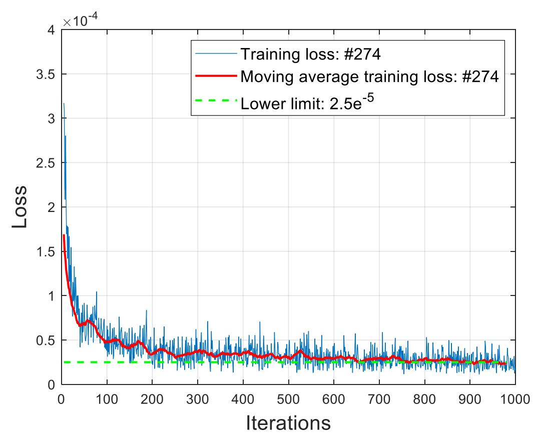
#405



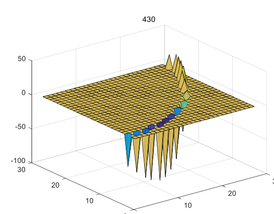
Validation error 5.6%



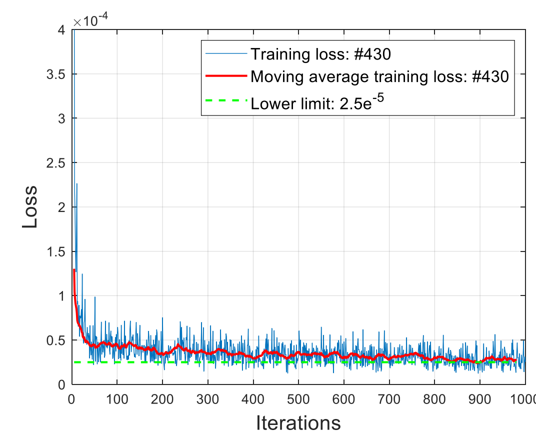
#274



Validation error 7.2%



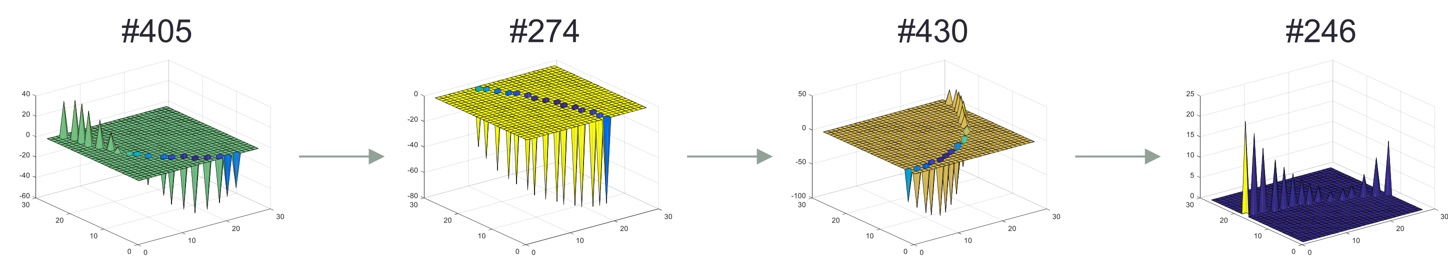
#430

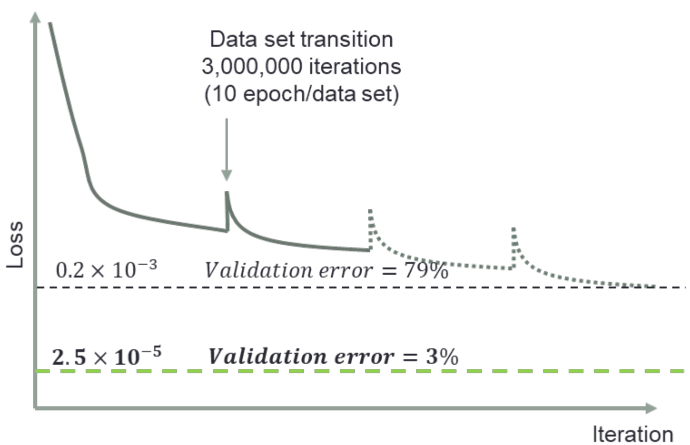


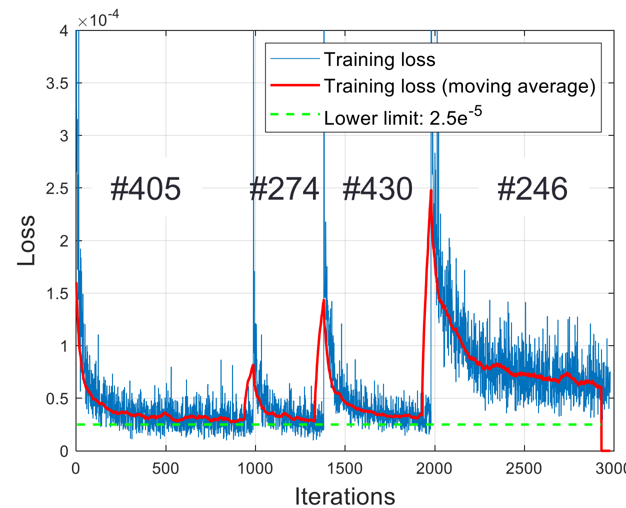
Validation error 3.2%

Appendix

- **Training two symmetric CNNs**
  - Training pre-trained CNNs with newly categorized patterns by ***transfer learning***
    - Perform training with selected dataset until the loss function reaches the target loss







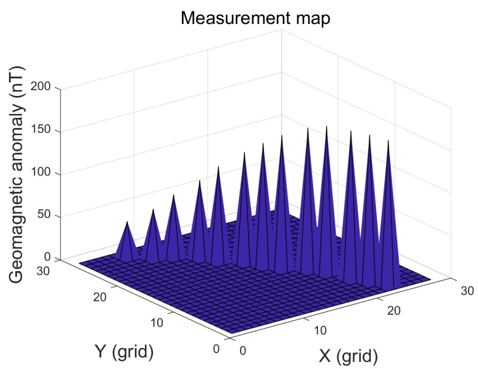
Random dataset selection

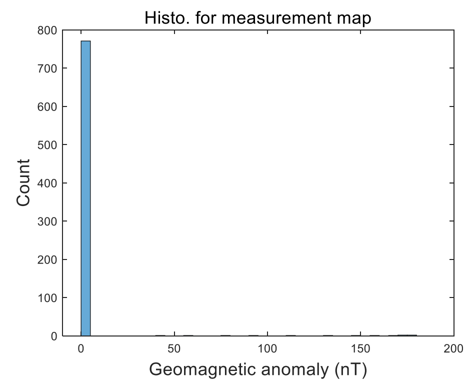
Sequential dataset selection

Appendix

- **Histogram for measurement map and search map**
  - Asymmetry between measurement map and search map
    - Measurement map is filled with zero where not having geomagnetic measurement

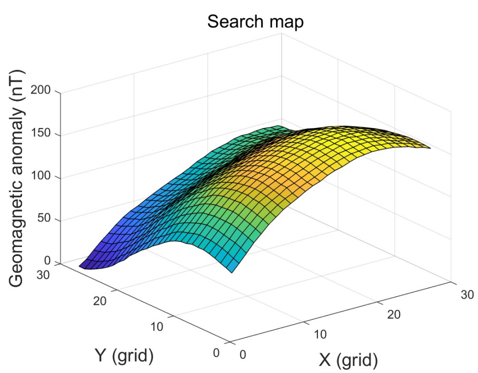
Measurement Pattern

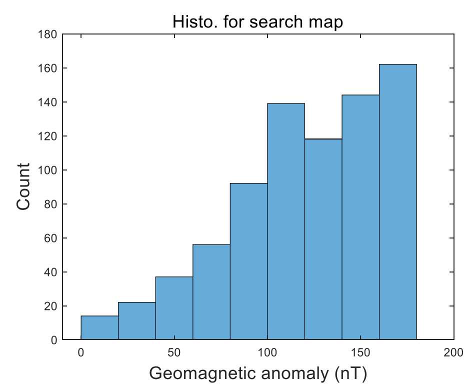




Asymmetry 96.5%

Search Map





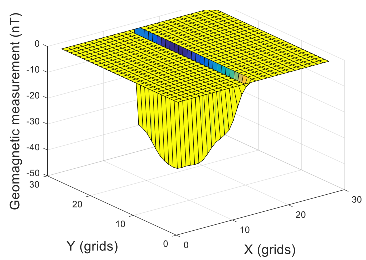
Map profile **Histogram**

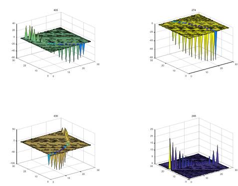
Geomagnetic matching for navigation

- **Characteristics of measurement pattern**

Single Pattern

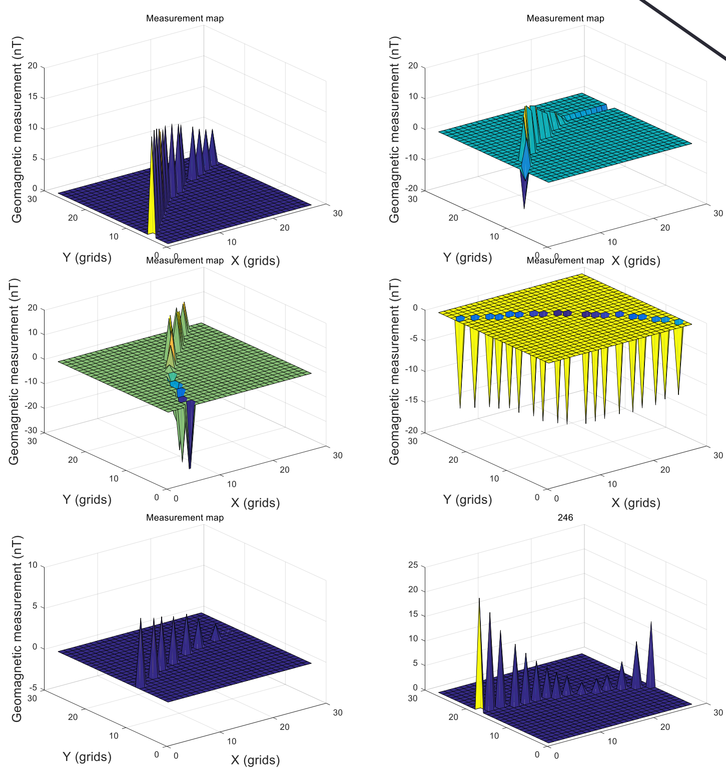
Measurement maps

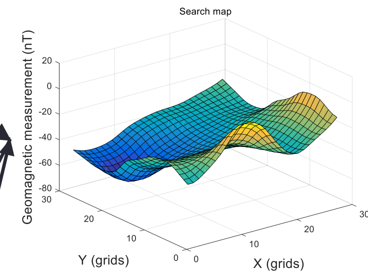




- The pattern of the measurements: fixed
- The numbers of measurements: fixed
- The value of the measurements: varied

Multiple Pattern





Search map

- *The pattern of the measurements: varied*
- *The numbers of measurements: varied*
- *The value of the measurements: varied*

➔ All randomization