



(12)

BREVET DE INVENȚIE

(21) Nr. cerere: **a 2021 00560**

(22) Data de depozit: **17/09/2021**

(45) Data publicării mențiunii acordării brevetului: **30/08/2023** BOPI nr. **8/2023**

(41) Data publicării cererii:
30/05/2022 BOPI nr. **5/2022**

(73) Titular:
• **UNIVERSITATEA TEHNICĂ DIN
CLUJ-NAPOCA, STR.MEMORANDUMULUI
NR.28, CLUJ-NAPOCA, CJ, RO**

(72) Inventatori:
• **POP MARIAN LEONTIN, NR.209A,
SAT VIILE, SEINI, MM, RO;**
• **TAMAS LEVENTE, STR.BISERICII
ORTODOXE, NR.18, CLUJ-NAPOCA, CJ,
RO**

(56) Documente din stadiul tehnicii:
**HE Y. Ș.A., "DEPTH ERRORS ANALYSIS
AND CORRECTION FOR TIME-OF-FLIGHT
(ToF) CAMERAS", SENSORS, NR. 92,
VOL. 71(1), PP. 1-18,
doi:10.3390/s17010092
www.mdpi.com/journal/sensors, 2017;**
**MARCO J. Ș.A., "DEEPTOF:
OFF-THE-SHELF REAL-TIME
CORRECTION OF MULTIPATH
INTERFERENCE IN TIME-OF-FLIGHT
IMAGING", ACM TRANSACTIONS ON
GRAPHICS, No.6, VOL. 36, ARTICLE 219,
PP. 219:1-21912,
https://dl.acm.org/doi/pdf/10.1145/3130800.
3130884, 2017; WO 2017160516 (A1)**

(54) **METODĂ PENTRU CORECȚIA SUPRAFETELOR PLANE
DIN IMAGINILE PROVENITE DE LA CAMERE
CU INFORMAȚIE DE DISTANȚĂ**



RO 135782 B1

1 Invenția se referă la o metodă de corecție, automată, a distorsiunilor datorate
efectului de interferență multiplă, din suprafețele plane, care poate să apară în imaginile
3 provenite de la camerele cu informație de distanță, camere cu senzori de tipul "Time-of-
Flight" (ToF).

5 Metoda cuprinde o componentă care folosește rețele neuronale convoluționale pentru
eliminarea artefactelor survenite în urma efectului de interferență multiplă, cât și o compo-
7 nentă pentru crearea setului de date necesar antrenării rețelei neuronale în metodele cunos-
cute, se încearcă eliminarea efectului de interferență multiplă din imaginile cu informație de
9 adâncime provenite de la camere ToF cu modulație cu undă continuă, în care se com-
pensează aceste erori folosind ecuațiile specifice pentru propagarea semnalului precum în
11 [1]-[3], [7].

13 Alte soluții, precum [1], [4], [5], [8], folosesc seturi de date sintetice pentru antrenarea
rețelelor neuronale, însă acestea nu produc rezultate foarte bune pentru datele reale, iar
15 metodele care folosesc date reale [5, 9] folosesc și senzori de înaltă precizie, mult mai
costisitori, pentru crearea setului de date.

17 Invenția diferă de cele menționate prin faptul că adresează problema eliminării
interferențelor multiple din imaginile de adâncime provenite de la camerele ToF emițătoare
de impulsuri folosind în același timp un set de date reale pentru antrenarea rețelei neuronale.

19 Problema tehnică pe care o rezolvă invenția este înlăturarea artefactelor efectului de
interferență multiplă din suprafețele plane, care apare în imaginile provenite de la camerele
21 cu informație de distanță, camere cu senzori de tipul ToF.

23 Se evidențiază, în primul rând, modul de creare al unui set de date reale folosind
procedee convenționale și fără a utiliza alți senzori adiționali, iar mai apoi modul de antrenare
al unei rețele neuronale folosind acest set de date pentru a putea înlătura artefactele
25 efectului de interferență multiplă.

27 Metoda de realizare a setului de date implică trei pași importanți:

29 1. Transformarea imaginii de adâncime 2D într-un set discret de puncte în spațiul
tridimensional, folosind parametrii intrinseci ai camerei.

31 2. Identificarea planurilor majore din norul de puncte 3D și proiectarea punctelor 3D
din fiecare plan distorsionat pe un plan ideal, rezultând planuri rectificate într-un mod iterativ.

33 3. Proiectarea puncte 3D înapoi în spațiul bidimensional al imaginii de adâncime
După translatare, folosim din funcționalitățile de bază prezente în biblioteca de
35 procesare a seturilor de puncte aflate în spațiul 3D denumită Point-Cloud Library (PCL),
astfel încât să proiectăm toate punctele care aparțin unei suprafețe planare pe un singur
plan, uniform, fără distorsiuni.

37 Noul set de puncte, astfel obținut, este reproiectat înapoi în spațiul bidimensional al
imaginii de adâncime. Imaginile de referință astfel obținute sunt servite rețelei neuronale
convoluționale, împreună cu imaginile originale, iar astfel rețeaua este capabilă să distingă
39 între cele două imagini, folosind o funcție de calculare a erorii (explicată mai jos), special
construită pentru acest tip de date, iar apoi să învețe ce anume trebuie să îmbunătățească
41 în imaginea de adâncime, fără a mai face anumite translatare în spațiul tridimensional

43 Metoda de corecție, automată, a distorsiunilor datorate efectului de interferență
multiplă, din suprafețele plane poate fi utilizată pentru orice tip de cameră și nu necesită
45 echipamente costisitoare pentru realizarea setului de imagini de referință necesare antrenării
rețelei neuronale.

47 Se dă în continuare un exemplu de realizare a invenției, în legătură cu fig. 1...4, care
reprezintă:

49 - fig. 1, structura sistemului;

- fig. 2, pașii necesari creării setului de date;

RO 135782 B1

| | |
|---|--|
| - fig. 3, pașii necesari antrenării rețelei neuronale; | 1 |
| - fig. 4, relația dintre punctele estimate de rețea și punctele de referință, în timpul procesului de antrenare. | 3 |
| Invenția se poate aplica la un sistem de forma celui prezentat în fig. 1. Sistemul este alcătuit dintr-un sistem ce calcul 1 , având implementat un program 2 , de corecție a suprafețelor plane achiziționate de la camera 3 , astfel încât imaginea 4 de adâncime este transformată în imaginea de adâncime 5 rectificată. | 5 7 |
| Camera 3 poate fi o cameră de tip Time-of-Flight emițătoare de impulsuri sau cu modulație cu undă continuă, și are rolul de a prelua imaginea unui spațiu 6 interior care conține suprafețe plane. | 9 |
| Programul 2 de corecție a suprafețelor plane constă dintr-o rețea neuronală 7 , care a fost în prealabil antrenată de un program 8 pe un set de date 9 . | 11 |
| Setul de date 10 a fost construit prin achiziționarea unui set de imagini de adâncime de la camera 3 , și procesarea acestor imagini de către programul 11 , pentru obținerea imaginilor 12 de adâncime de referință. | 13 15 |
| Metoda, conform invenției, presupune, realizarea următoarelor etape: | |
| - Preluarea imaginii cu ajutorul camerei și crearea setului de date necesare antrenării rețelei neuronale | 17 |
| - Realizarea modelului neuronal; | 19 |
| - Exploatarea modelului neuronal pentru corecția erorilor; | |
| În prima etapă, pentru crearea setului de date necesar antrenării rețelelor neuronale, s-au achiziționat, folosind camera ToF, o serie de imagini de adâncime ale unor spații interioare care conțin una sau mai multe suprafețe planare, cum ar fi podea, tavan, pereți, etc. Imaginile astfel achiziționate sunt transformate, folosind parametrii intrinseci ai camerei, într-un nor de puncte, în spațiul tridimensional. În spațiul 3D, am folosit funcții de bază din biblioteca Point Cloud Library (PCL), pentru a putea identifica și extrage toate punctele care aparțin planului major din încăpere (e.g. podeaua), dar în două iterații, deoarece funcțiile prezente în biblioteca PCL nu pot să identifice și să extragă toate punctele dintr-un plan atât de distorsionat. Astfel, în prima iterație am folosit funcțiile din bibliotecă pentru a identifica un set mai mic de puncte ce aparțin planului major, dar care nu sunt afectate de efectul de interferență multiplă (de ex. punctele care sunt mai apropiate de cameră) și am salvat coeficienții planului din care fac parte. În cea de a doua iterație, am folosit aceleași funcții din biblioteca PCL, dar cu un set de parametri care să permită extragerea tuturor punctelor din planul major. Astfel, punctele selectate în cea de a doua iterație sunt proiectate pe planul cu coeficienții extrași în prima iterație, obținând astfel un plan rectificat, fără distorsiuni. După rectificarea planurilor majore în spațiul 3D, norul de puncte este translatat înapoi într-o imagine 2D, obținând astfel imaginile de referință pentru antrenarea modelului. | 21 23 25 27 29 31 33 35 37 |
| În cea de a doua etapă, am utilizat un model deja existent, Feature Pyramid Network (FPN), care oferă rezultate bune în procesele de segmentare a imaginilor RGB, în sarcinile necesare identificării și clasificării obiectelor prezente în imagini, la nivel de pixel. Din acest motiv am ales acest model, pe care l-am modificat astfel încât să proceseze imagini de adâncime, modificându-i astfel și funcția de calcul a erorii pentru a putea servi obiectivului nostru de rectificare a suprafețelor plane. | 39 41 43 |
| În ultima etapă, modelul neuronal antrenat este folosit de către un program care preia imaginile de la camera ToF și le servește pe intrarea modelului care mai apoi le procesează și oferă la ieșire o imaginile cu planurile rectificate din imagine. | 45 |

RO 135782 B1

1 În fig. 2 se prezintă organigrama pașilor necesari generării setului de date necesar
antrenării și exploatării modelului neuronal, în care se pune în evidență modul de procesare
3 a punctelor în spațiul tridimensional folosind biblioteca PCL. Ideea principală din această
etapă constă în folosirea algoritmului RANSAC, a bibliotecii, de două ori: prima dată pentru
5 a extrage coordonatele planului care aproximează cel mai bine planul din imagine, iar a doua
oară pentru a extrage cât mai multe puncte care aparțin planului major din imagine. Setul de
7 puncte rezultat în urma celei de-a doua iterații a algoritmului este proiectat pe planul ce are
coordonatele determinate în prima iterație, rezultând astfel un plan ideal care nu conține
9 artefacte.

Blocurile de calcul necesare generării setului de date din fig. 1 sunt:

11 Blocul **13** efectuează preluarea datelor de adâncime de la camera **3**. În acest bloc,
fluxul de imagini achiziționat de la cameră prin intermediul ROS (Robot Operating System)
13 este împărțit în cadrele componente ale acestuia, salvat sub formă de imagini.

15 Blocul **14** are rolul de translatare a punctelor din spațiul 2D în spațiul 3D, folosind
parametrii intrinseci ai camerei. Translatarea se face cunoscând matricea intrinsecă a
camerei, care conține distanța focală a camerei, precum și centrul senzorului camerei **3**,
17 folosind o funcție din biblioteca PCL.

19 Blocul **15**, are rolul de a prelua norul de puncte obținut în blocul anterior, pentru a
identifica, folosind algoritmul RANSAC, implementat în biblioteca PCL, toate planurile majore
existente și de a procesa fiecare plan în parte, astfel încât să identifice un set de puncte care
21 aparțin unui plan ideal ce aproximează cel mai bine planul curent, care conține artefacte.
Selectarea punctelor care aparțin unui plan ideal se realizează prin setarea unor
23 constrângeri mai mari în algoritmul de identificare al planurilor, prin setarea corespunzătoare
a parametrilor acestuia

25 Blocul **16**, este aproape identic cu blocul **15**, doar că de această dată se folosesc
parametrii cu constrângeri mai slabe, care îi permit algoritmului să selecteze întregul plan
27 cu toate punctele sale, chiar dacă acesta conține și artefacte.

29 Blocul **17**, preia toate punctele selectate în blocul **16** și le proiectează pe planul ideal,
ce are coordonatele extrase în blocul **15**.

31 Blocul **18** realizează înlocuirea punctelor din blocul **16**, adică punctele planului ce
conține artefacte, cu punctele din blocul **17**, adică planul recitificat, lăsând restul punctelor
care nu aparțin planului neatins.

33 Blocul **19**, realizează proiectarea norului de puncte înapoi în imaginea de adâncime,
folosind tot parametrii intrinseci ai camerei **3**, dar folosind o funcție inversă din librăria PCL.

35 În fig. 3, sunt prezentați pașii necesari pentru antrenarea rețelei neuronale
convolutionale, în care am ales un model de tipul FPN [6]. După ce am încărcat în model
37 perechile de imagini de adâncime folosim o funcție personalizată de calculare a erorii dintre
imaginea prezisă de către rețea și imaginea de referință, în spațiul 3D. Fiindcă artefactele
39 pe care dorim să le înlăturăm se observă într-un mod mai pronunțat în spațiul tridimensional
am ales să folosim o rețea convoluțională care este capabilă să obțină rezultate bune în
41 procesarea de imagini, dar pentru care calculăm o eroare în spațiul 3D.

Blocurile de calcul necesare antrenarea rețelei neuronale din fig. 3 sunt:

43 Blocul **20** are rolul de încărcare în programul **9** a imaginilor **4** de adâncime de la
camera **3**, împreună cu imaginile **10** de adâncime de referință pentru a putea fi servite pe
45 intrarea rețelei neuronale **7**.

RO 135782 B1

Blocul **21** are rolul de a normaliza imaginile în intervalul [0, 1] , pentru a optimiza procesul de antrenare al rețelei **7**. 1

Blocul **22** inițializează modelul **8** prin încărcarea acestuia în memoria sistemului de calcul **4**. 3

Blocul **23** preia imaginea de pe ieșirea modelului neuronal ca răspuns la imaginile de intrare și o trimite în funcția de calculare a erorii, din blocul **24**. 5

Blocul **24** are rolul de a prelua imaginea de ieșire de la blocul **23**, împreună cu imaginea de referință încărcată în blocul **20** și translatarea lor în spațiul tridimensional. Odată realizată translatarea, se calculează eroarea dintre cele două seturi de puncte, pentru a putea determina cât de departe sau cât de aproape sunt punctele din imaginea de referință față de cele din imaginea estimată de rețea. 7
9
11

Blocul **25** are rolul de a actualiza parametrii interni ai modelului neuronal **8**, în funcție de eroarea estimată în blocul **24**, după care procesul se reia de la blocul **23** până când eroarea calculată este suficient de mică. 13

În fig. 4 se prezintă relația dintre punctele din planul estimat de către rețea și punctele din planul de referință. Practic, în calcularea erorii se calculează într-o primă fază o radicalul unei erori medii pătratice logaritmice realizate între fiecare punct din imaginea estimată de rețea și imaginea de referință, denumită *RMSE_log*, iar mai apoi se calculează și câte o eroare medie pătratică a punctelor, în funcție de coordonata X, coordonat Y și coordonata Z, denumite *lossX*, *lossY*, respectiv *lossZ*. Aceste patru ecuații sunt mai apoi combinate pentru a obține o ecuație generală de calculare a erorii 15
17
19
21

Mai jos este prezentată formula folosită pentru calcularea erorii: 23

$$lossX = \frac{1}{n} \sum_{i=1}^n (x_{P_i} - x_{P_i^*})^2 \quad 25$$

$$lossY = \frac{1}{n} \sum_{i=1}^n (y_{P_i} - y_{P_i^*})^2 \quad 27$$

$$lossZ = \frac{1}{n} \sum_{i=1}^n (z_{P_i} - z_{P_i^*})^2 \quad 29$$

$$RMSE_log = \sqrt{\sum_{i=1}^n (\ln|z_{P_i}| - \ln|z_{P_i^*}|)^2} \quad 31$$

unde: 33

- x_{P_i} , y_{P_i} , z_{P_i} sunt coordonatele spațiale ale punctului P_i din imaginea de referință;

- $x_{P_i^*}$, $y_{P_i^*}$, $z_{P_i^*}$ sunt coordonatele spațiale ale punctului P_i^* din imaginea estimată de rețeaua neuronală 35

- n este numărul total de puncte din imaginea de referință. 37

Folosind ecuațiile de mai sus se ajunge la forma finală a erorii: 39

$$\varepsilon = 10RMSE_log(3 - e^{lossX} - e^{lossY} - e^{lossZ}) \quad 41$$

Ideea inovatoare iese în evidență prin faptul că metoda este capabilă să corecteze erorile sistematice de la camerele de tip ToF în timp real chiar și pe dispozitive înglobate cu procesor GPU folosind tehnici de inteligență artificială. 43
45

1 Bibliografie

- 3 [1] Guo, Q., Frosio, I., Gallo, O., Zickler, T., & Kautz, J. (2018). *Tackling 3d tof*
5 *artifacts through learning and the flat dataset in Proceedings of the European Conference on*
6 *Computer Vision (ECCV)*, pp. 368-383.
- 7 [2] Jimenez, D., Pizarro, D., Mazo, M., & Palazuelos, S. (2014). *Modeling and*
8 *correction of multipath interference in time of flight cameras*. *Image and Vision Computing*,
9 32(1), 1-13.
- 10 [3] US10215856B1.
- 11 [4] Marco, J., Hernandez, Q., Munoz, A., Dong, Y., Jarabo, A., Kim, M. Gutierrez, D.
12 (2017). *Deeptof: off-the-shelf real-time correction of multipath interference in time-of-flight*
13 *imaging*. *ACM Transactions on Graphics (ToG)*, 36(6), 1-12.
- 14 [5] He, Y., Liang, B., Zou, Y., He, J., & Yang, J. (2017). *Depth errors analysis and*
15 *correction for time-of-flight (ToF) cameras*. *Sensors*, 17(1), 92.
- 16 [6] Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S., 2017.
17 *Feature pyramid networks for object detection*. In *Proceedings of the IEEE conference on*
18 *computer vision and pattern recognition*, pp. 2117-2125.
- 19 [7] Fuchs, S., 2010, *Multipath interference compensation in time-of-flight camera*
20 *images*. In *2010 20th International Conference on Pattern Recognition* (pp. 3583-3586).
21 IEEE.
- 22 [8] Agresti, G. and Zanuttigh, P., 2018. *Deep learning for multi-path error removal in*
23 *ToF sensors*. In *Proceedings of the European Conference on Computer Vision (ECCV)*
24 *Workshops*.
- 25 [9] Son, K., Liu, M.Y. and Taguchi, Y., 2016 *Learning to remove multipath distortions*
in time-of-flight range images for a robotic arm setup in 2016 IEEE International Conference
on Robotics and Automation (ICRA), pp. 3390-3397

RO 135782 B1

Revendicare

| | |
|---|----|
| | 1 |
| Metodă pentru corecția suprafețelor din imaginile provenite de la camere cu informație de distanță folosind rețele neuronale convoluționale de tip Feature Pyramid Network (FPN), implementată pe un sistem de calcul (1), având implementat un program (2) de corecție a suprafețelor plane achiziționate de la o cameră (3), caracterizată prin aceea că pentru rectificarea imaginilor se parcurg următoarele etape: | 3 |
| | 5 |
| | 7 |
| - achiziția cu ajutorul camerei (3) unei serii de imagini de adâncime ale unor spații interioare care conțin una sau mai multe suprafețe planare, cum ar fi podea, tavan sau pereți; | 9 |
| - transformarea imaginii de adâncime 2D într-un set discret de puncte în spațiul tridimensional, folosind parametrii intrinseci ai camerei (3) pentru crearea setului de date necesare antrenării rețelei neuronale; | 11 |
| - identificarea planurilor majore din norul de puncte 3D prin iterații multiple și proiectarea punctelor 3D din fiecare plan distorsionat pe un plan ideal, rezultând planuri rectificate; | 13 |
| | 15 |
| - folosind planurile majore rectificate (corectate) norul de puncte este translatat înapoi într-o imagine 2D, obținând astfel imaginile de referință pentru antrenarea modelului; | 17 |
| - realizarea modelului neuronal folosind modelul Feature Pyramid Network (FPN), cu funcția de calculare a erorii adaptată pentru a putea servi la rectificarea suprafețelor plane; | 19 |
| - calculul erorilor în spațiul 3D dintre imaginile cu planurile rectificate estimate de către rețeaua neuronală și imaginile de referință din setul de date; | 21 |
| - proiectarea punctelor 3D înapoi în spațiul bidimensional al imaginii de adâncime. | |

(51) Int.Cl.
G06V 10/24 (2022.01);
G06T 7/60 (2006.01);
G06T 1/40 (2006.01)

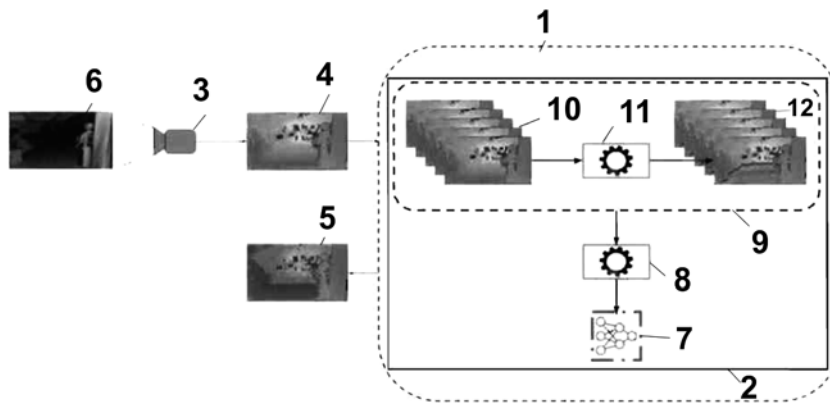


Fig. 1

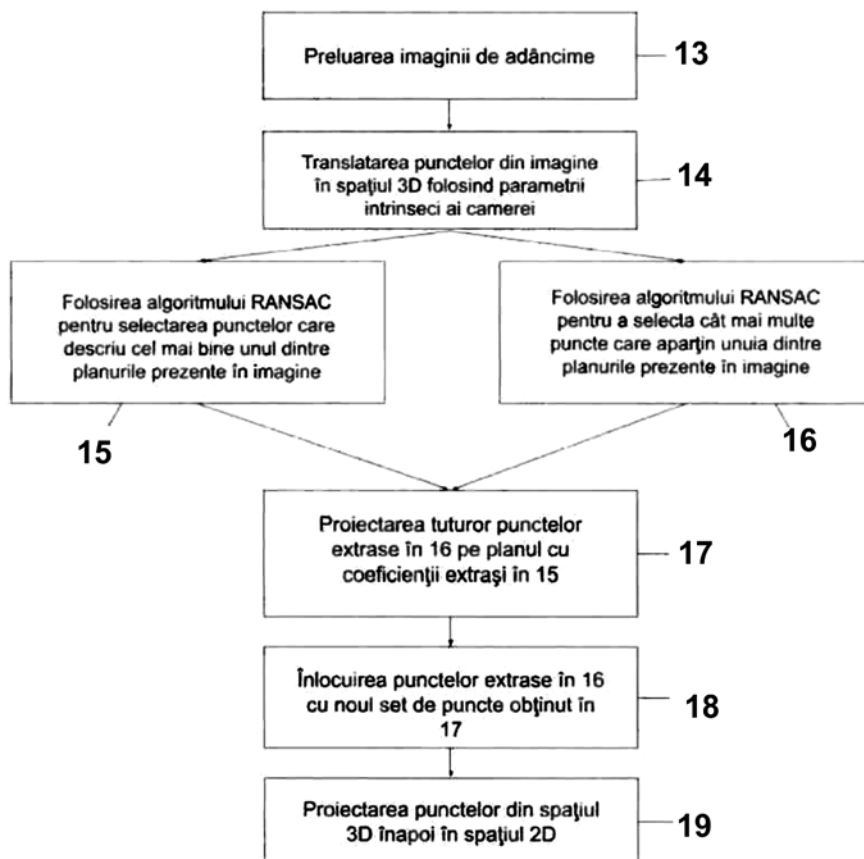


Fig. 2

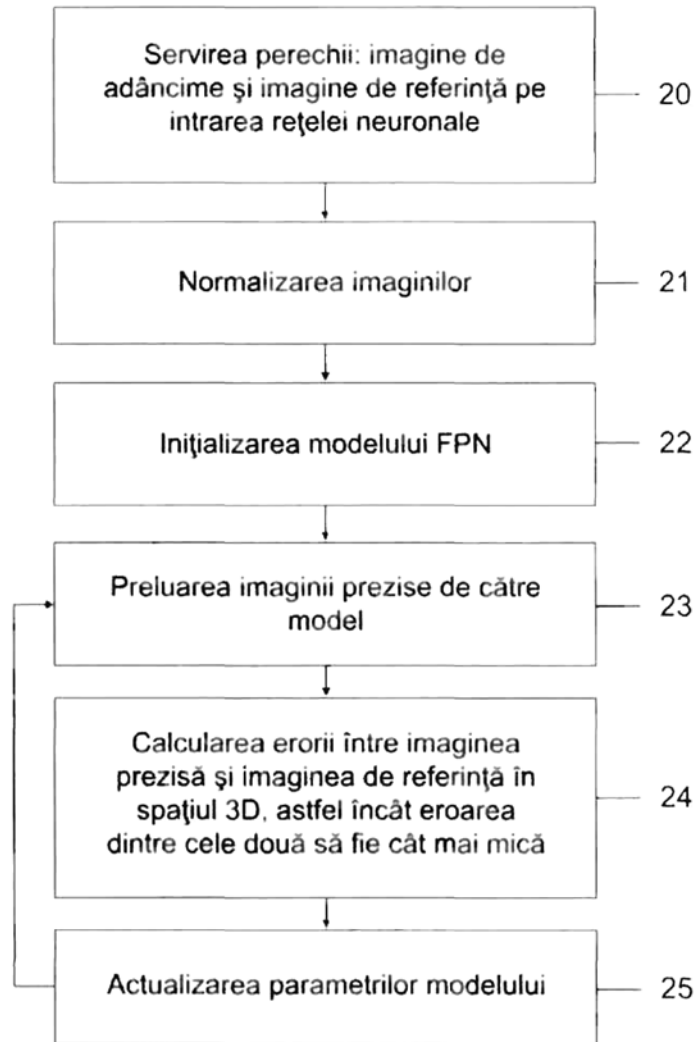


Fig. 3

(51) Int.Cl.

G06V 10/24^(2022.01);

G06T 7/60^(2006.01);

G06T 1/40^(2006.01)

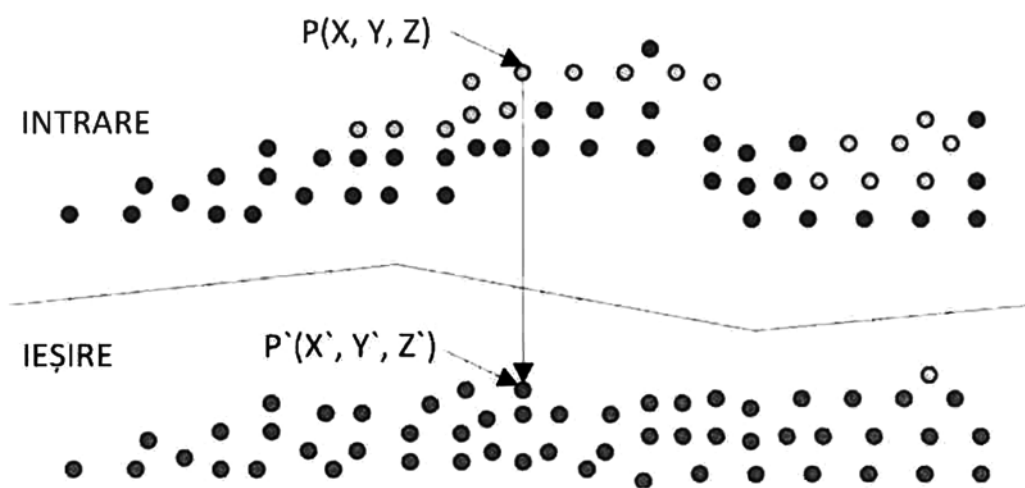


Fig. 4

