

Neurális Háló Alapú Hibakereső Algoritmus Tervezése Agilis Gyártáshoz

Neural Network Based Error Detection Algorithm in Agile Manufacturing

Biebel Botond^{1, 2}, Bilszky Márk^{1, 3}, Dr. Czmerk András^{1, 4}

¹Budapesti Műszaki és Gazdaságtudományi Egyetem, 1111 Budapest, Mechatronika, Optika és Gépészeti Informatika Tanszék, Budapest Bertalan Lajos u. 4-6 D. épület, 4. emelet, 407, 1111

²e-mail: botond.biebel@edu.bme.hu

³e-mail: mark.bilszky@edu.bme.hu

⁴e-mail: czmerk@mogi.bme.hu

Abstract

Modern consumer demands require a broader and more diverse range of products, leading to the growing adoption of agile manufacturing technologies. However, the production of unique items introduces significant challenges, as traditional quality control methods are often ineffective at detecting one-off defects. This research seeks to develop an economical solution by leveraging CAD models to generate reference images of defective components. These images will be used to train a neural network for detecting faults, which can then be implemented in real-time quality control using an integrated camera system within the manufacturing environment. The study focuses on simulating production errors in a virtual environment, to create an automated system capable of recognizing individual defects. This approach minimizes human intervention and remains adaptable to various components, provided their CAD models and known defect characteristics are available.

Keywords: neural network, computer vision, agile manufacturing, synthetic data generation, error detection

Kivonat

A fogyasztói igények egyre változatosabb termékskálát igényelnek, melynek következtében a modern gyártási rendszerekben egyre inkább elterjedtek az agilis gyártási technológiák. Az egyedi termékek gyártása azonban komoly kihívásokat jelent, mivel a hagyományos minőségellenőrzési módszerek gyakran nem elég hatékonyak az egyedi hibák felismerésében. A dolgozat célja egy költséghatékony megoldás kidolgozása, amely CAD-modellek felhasználásával hoz létre referencia képeket hibás alkatrészekről. Ezek a képek később egy neurális háló betanítására szolgálnak, amelyet a gyártási folyamatban valós idejű minőségellenőrzésre lehet alkalmazni integrált kamerarendszer segítségével.

A kutatás a gyártási hibák virtuális környezetben történő szimulálására összpontosít. A munkánk eredményeként egy automatizált rendszert alakítunk ki a jellemzően előforduló hibák vizsgálatához. Eljárásunk minimalizálja az emberi beavatkozást, és alkalmas különböző alkatrészek minőségellenőrzésére, feltéve, hogy rendelkezésre állnak a CAD-modellek és a hozzájuk kapcsolódó jellegzetes hibák.

Kulcsszavak: agilis gyártás, neurális háló, minőségellenőrzés, CAD-modellek, hibák szimulációja, szintetikus adat létrehozás

1. BEVEZETÉS

A gyártási hibák automatikus ellenőrzése egy gyakran felmerülő kihívás, amely jelentős költségmegtakarítást és minőségjavulást eredményezhet az ipari folyamatokban. Jelenleg számos bevált tesztelési módszer létezik a hagyományos gyártási eljárások során. Néhány példa ezek közül [1]: ultrahangos tesztelés, mágneses részecskés tesztelés (MT), ozmózis tesztelés, röntgenes tesztelés.

A gépi látásra épülő hibafelismerési eljárások gyakran alkalmaznak mélytanulási algoritmusokat, mint például konvolúciós neurális hálókat (CNN), autoencoder és rekurrens neurális hálókat (RNN) [1]. Ezek az algoritmusok nagy pontossággal képesek azonosítani a hibákat, és a tanítási fázis után viszonylag alacsony számítási kapacitást igényelnek, mivel ekkor már csak a súlymátrixokkal végzett lineáris műveletek szükségesek [2]. Ez különösen előnyös valós idejű ellenőrzési feladatok esetén, például gyártósorok felügyeleténél.

A mélytanulási modellek hatékony működéséhez azonban nagy mennyiségű tanítóadat szükséges, amelynek előkészítése és annotálása jelentős idő- és munkaerőigénnyel jár. Az adatkomplexitás gyakran exponenciálisan nő a jellemzők számával, továbbá a valós gyártási hibák ritkasága és szimulálásuk költsége is nehezíti a modellfejlesztést.

Az agilis gyártás egy modern stratégia, amely a gyártási folyamatok rugalmasságát hangsúlyozza a gyors piaci alkalmazkodás érdekében [3]. Azonban a változó gyártási környezet és a kis szériás termelés miatt a hagyományos adatgyűjtési módszerek gyakran nem elég gyorsak. Ennek megoldására a szintetikus adatok generálása CAD modellek és 3D szimulációk révén egy hatékony alternatíva.

Ez a megközelítés csökkenti az annotációs idő- és költségigényt, miközben lehetőséget nyújt ritkán előforduló hibák modellezésére. A szimulációval továbbá a környezeti tényezők, például a fényviszonyok és textúrák könnyen variálhatók, ami javítja a modell általánosítási képességét és alkalmazhatóságát különböző gyártási környezetekben.

A kutatás célja a mesterséges intelligencia alkalmazási lehetőségeinek bővítése a tömeggyártás területén, különös tekintettel az agilis gyártási elvekre. Ennek érdekében egy olyan rendszert fejlesztettünk ki, amely neurális hálók és virtuális szimulációk segítségével hatékonyan képes azonosítani és kiszűrni a hibás alkatrészeket. A javasolt megközelítés célja, hogy minimalizálja az emberi beavatkozást, miközben növeli a minőségellenőrzési folyamatok gyorsaságát és pontosságát, ezáltal hozzájárulva a gyártási folyamatok optimalizálásához.

2. KÉP GENERÁLÁS

A neurális hálózatokon alapuló hibakereső algoritmusunk fejlesztésének egyik kulcsfontosságú eleme a megfelelő adatok létrehozása. Ez a fejezet ismerteti a szintetikus adatgenerálási módszereinket, amelyek lehetővé teszik a nagy mennyiségű, jó minőségű képi adatok automatizált létrehozását a neurális hálók hatékony tanítása érdekében.

2.1. Szintetikus adatgenerálás

A neurális hálózatok hatékony tanításához elengedhetetlen a nagy mennyiségű és változatos adathalmaz. Ez különösen fontos a képfelismerési és osztályozási feladatok esetében. Az adatok manuális előállítása vagy begyűjtése azonban nemcsak költséges, hanem időigényes folyamat is [4], ráadásul az adatigény a feladat komplexitásával és a képek mennyiségének növekedésével együtt nő. Erre a kihívásra kínál hatékony megoldást a szintetikus adatgenerálás, amely az automatizált megoldásoknak köszönhetően jelentősen csökkenti az adatgyűjtés költségeit és időigényét [4].

A szintetikus adatgenerálás során a képi adatok mesterségesen vannak előállítva, törekedve arra, hogy azok vizuálisan és statisztikai szempontból is hasonló tulajdonságokkal rendelkezzenek, mint a valós adatok. Ez lehetővé teszi a modellek hatékony tanítását. További előnye ennek a megközelítésnek, hogy a szintetikus adatgeneráló algoritmusok elkészítése után az adatok előállítása egyszerűen, lineáris időben skálázható, biztosítva a nagy mennyiségű, konzisztens adatkészlet gyors előállítását. Az agilis gyártás környezetében, ahol a gyártott termékek gyakran változnak és kis szériában készülnek, a szintetikus adat alapú megközelítés különösen értékes, mivel lehetővé teszi a gyors adaptációt az új termékekhez és a ritkán előforduló hibatípusok modellezését.

2.2. Szintetikus adatgenerálási eljárások

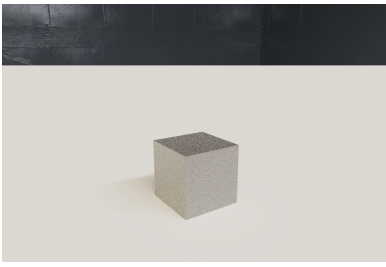
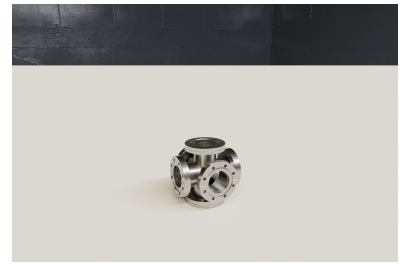
A szintetikus adatgenerálás három fő típusra osztható, amelyek különböző alkalmazási lehetőségeket kínálnak az adatbázis gazdagítására és a neurális hálók tanítására [5]. A teljesen szintetikus adatok teljes mértékben mesterségesen generálódnak, míg a részlegesen szintetikus adatok esetében a valós adatokat mesterségesen generált elemekkel kombinálják. A harmadik típus a Generative Adversarial Networks (GAN-ek) használata, amely képes rendkívül realisztikus adatokat generálni, amelyek jól illeszkednek a valós adatmintákhoz, viszont számítási igényük magas.

Jelen kutatásban egy teljesen szintetikus adatgenerálási eljárást alkalmazunk. Az eljárás során egy saját fejlesztésű, Blender alapú algoritmust használunk, amely lehetővé teszi a különböző alkatrészek és hibatípusok rugalmas modellezését, valamint a környezeti tényezők pontos beállítását.

2.3. Adatgenerálási módszertan

2.3.1. Geometriai modell elkészítése

Az adatgenerálás első lépése a geometria 3D modelljének elkészítése, amelyet STL formátumba konvertálunk. A projekt során három modellt használunk: két alapeometriát (kocka (1.ábra) és gömb (2.ábra)) és egy összetettebb, hat ágú csőidomot (3.ábra). Az egyszerű geometriák lehetőséget adnak a gyors betanulásra és tesztelésre, míg a komplex forma demonstrálja az eljárás alkalmazhatóságát bonyolultabb alkatrészekben is.

1. ábra. *Kocka*2. ábra. *Gömb*3. ábra. *Hat ágú cső idom*

2.3.2. Jellegzetes hibák meghatározása

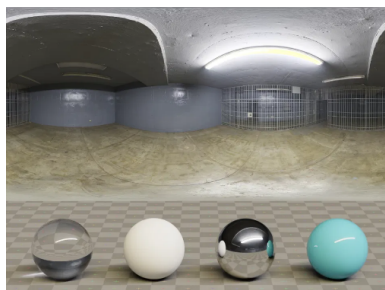
Az algoritmus fejlesztésének elengedhetetlen része a jellemző hibák meghatározása. Ezek a hibák lehetnek bármilyen gyártásból, megmunkálásból következő, előre jól definiált és számszerűsíthető típushibák. Gyakori típushibáknak számítanak például a mérethibák, tűréshibák, felületi egyenlőtlenségek vagy az illesztési hibák [6]. Adatgenerálás során két fő hibatípust alkalmazunk: élettörést (4.ábra) és V alakú kimetszést (5.ábra). Az élettörés gyakran előfordul a mechanikai terhelésnek kitett éles sarkoknál, ahol a feszültség koncentrációk miatt az anyag gyengül. A V alakú kimetszés olyan hibatípus, amely nem megfelelő beállítások vagy kopott szerszámok használata miatt keletkezik, és a felület egyenetlenségét okozza, különösen precíziós alkatrészeknél.

4. ábra. *Generált élettörés*5. ábra. *Generált V alakú kimetszés*

2.3.3. Környezeti tényezők beállítása

A környezeti tényezők pontos modellezése kritikus a valósághű adatok előállításához. Az alkatrész anyagának modellezése az egyik legfontosabb paraméter, amely befolyásolja többek között a fényvisszaverődési tulajdonságokat. A használt textúra egyedi fejlesztés, amely Fused Deposition Modeling (FDM) technológiával nyomtatott felületet modellez különböző rétegekből és szabálytalan struktúrákból álló anyagmodellel.

A fényviszonyok modellezésére High Dynamic Range Image (HDRI) térképet alkalmazunk, amely irodai világítást szimulál [7] (6.ábra). Ez lehetővé teszi a beltéri fényviszonyok pontos reprodukálását, így a szintetikus képek olyan környezetben készülnek, amelyet később a valóságban fogunk tudni reprodukálni.

6. ábra. *HDRI térkép*

2.3.4. Blender kód

A képgenerálás skálázhatósága és egyszerű futtathatósága érdekében a teljes modellezési folyamatot a Blender Python IDE-n keresztül végezzük. A Blender Python API pedig lehetővé teszi számunkra, hogy egyetlen

szkript segítségével bármilyen számú képet generáljunk, különböző hibákkal, fénytérképekkel és textúrákkal. Ezzel a megoldással lineáris időbeli komplexitással, gyorsan és hatékonyan generálhatók a szintetikus képek, amelyek később felhasználhatók a neurális háló tanítására.

3. NEURÁLIS HÁLÓZAT FEJLESZTÉSE

3.1. Konvolúciós neurális háló

Az alkalmazott konvolúciós neurális háló (CNN) hatékonyan támogatja a hibafelismerési feladatokat azáltal, hogy absztraktabb jellemzőket képes kinyerni a teljesen összekapcsolt hálózatokhoz képest. A konvolúciós és pooling rétegek csökkentik a képek komplexitását, megőrizve a legfontosabb vizuális jellemzőket, miközben a hálózat robusztussá válik az elhelyezkedési és orientációs változásokkal szemben.

Ez a megközelítés mérsékli a modell komplexitását és az adatigényt, mivel a CNN az objektumokat függetlenül felismeri azok pozíciójától. A hierarchikus rétegstruktúra mélyebb reprezentációkat tesz lehetővé, javítva a hibafelismerés pontosságát. A hálózat implementációját a TensorFlow Keras [8] keretrendszerrel végeztük, amely gyors és átlátható modellezést biztosít.

A hálózat három konvolúciós és négy teljesen összekapcsolt (dense) rétegből áll. A konvolúciós rétegek kernel mérete fokozatosan csökken, mindegyiket maxpooling (jellemzőkiemelés, méretcsökkentés) követi a dimenziók csökkentése érdekében. Az utolsó konvolúciós réteg után egy Global Average Pooling (átlagolás, jellemzőösszegzés) réteg található.

A fully connected rétegek absztrakt összefüggéseket keresnek, aktivációs függvényként rectified linear unit (ReLU) függvényt alkalmaztunk, mely a negatív értékeket nullára állítja, míg a pozitív értékeket változatlanul hagyja, segítve a hálózatot a nemlineáris kapcsolatok hatékonyabb modellezésében.

A végső réteg Softmax aktivációt használ, mivel a feladat többosztályos klasszifikáció. A Softmax függvény minden kimeneti neuronhoz valószínűségi értéket rendel, amely meghatározza a bemenet osztályát. A kimeneti értékek összege 1, így azok valószínűségi eloszlásként értelmezhetők.

3.2. Transferlearning

Eljárásunk során kezdetben egyetlen hibatípusra tanítunk egy CNN segítségével. Ezt követően a modell továbbfejlesztésére került sor, hogy képes legyen az új, akár később felmerülő hibatípusok felismerésére is. A TL módszer alkalmazása lehetővé teszi számunkra, hogy kihasználjuk a korábban betanított modellek által megszerzett tudást, így az új feladatra történő tanítás gyorsabb és erőforrás-kímélőbb módon valósulhat meg.

A konvolúciós rétegek lefagyasztása után az új hálózat az eredeti CNN hálózattal megegyező dense rétegeket tartalmaz. Az optimalizáláshoz Stochastic Gradient Descent (SGD)¹ optimalizálót alkalmazunk [9] ciklikus tanulási rátával, miközben a dense rétegek között dropout-ot használunk a túlillesztés csökkentésére. Az új rétegek közé batch normalization-t is beillesztünk a gyorsabb és stabilabb tanulás érdekében. A modell finomhangolása (fine-tuning) lehetővé teszi, hogy az új hibák felismeréséhez a hálózat jobban alkalmazkodjon.

4. EREDMÉNYEK KIÉRTÉKELÉSE

4.1. Adatgenerálás

A kezdeti paraméterek megadása után a folyamat teljesen automatizált módon zajlik le, óránként nagyságrendileg 1000 képet tudunk generálni. Egy másik gépen tesztelve az eredmények azt mutatják, hogy a hardver minimális javítása, például egy erősebb GPU használata jelentősen növelheti a képgenerálás sebességét, és így a folyamatot jól skálázhatóvá teszi. Ez különösen fontos, mivel a szintetikus adatgenerálás a gépi tanulási modellek számára szükséges nagy mennyiségű adat előállításához alapvető lépés. A gyorsabb generálás tehát nemcsak a tanulási folyamat hatékonyságát javítja, hanem időt is megtakarít a kutatási és fejlesztési ciklusok során, ideálissá téve az agilis gyártási rendszerekben való alkalmazásra.

¹Egy nemlineáris optimalizációs probléma megoldása során a célfüggvény leegyszerűsítésének hatékony módszere a lokális lineáris közelítés. Ezt az eljárást követve az Stochastic Gradient Descent (SGD) módszer a bonyolult függvények minimumát lépésről-lépésre, iteratíván közelíti, a gradiens irányában történő korrekciókkal.

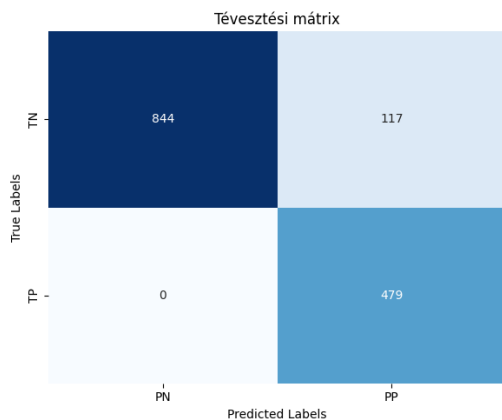
4.2. Neurális hálózatok kiértékelése

4.2.1. Konvolúciós Neurális Háló (CNN)

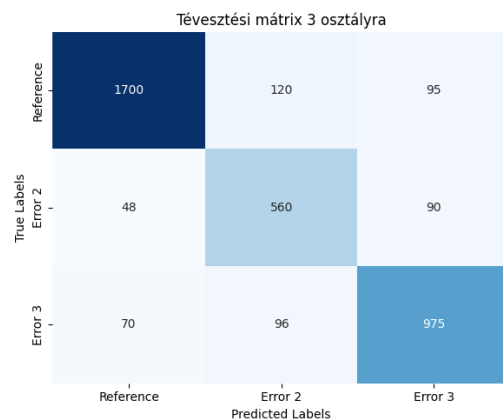
- **Találati arány (Accuracy):** A teszt találati aránya 92%. A magas találati arány azt mutatja, hogy a modell nagy pontossággal osztályoz. (7. ábra)
- **Pontosság (Precision):** 80%, jól osztályozott az adott osztályokban kevés hamis pozitívval. Mivel ez az érték is magas, a háló nem tanult túl. (7. ábra)
- **Tanítási idő:** Hozzávetőlegesen 41 perc

4.2.2. Transfer Learning

- **Találati arány (Accuracy):** A teszt találati aránya 78%, a magas találati arány azt mutatja, hogy a modell nagy pontossággal osztályoz (8. ábra).
- **Pontosság (Precision):** A pontosság magas értéke, 77% tükrözi, hogy a tanulás sikeres volt (8. ábra).
- **Tanítási idő:** Hozzávetőlegesen 16 perc



7. ábra. CNN tévesztési mátrixa



8. ábra. TL tévesztési mátrixa

4.2.3. Modellek összehasonlítása

A TL architektúra alkalmazása jelentősen csökkenti az új osztály hozzáadásának tanítási idejét. Ha először egy CNN modellt használunk egy hibatípus azonosítására, majd ezt követően alkalmazzuk a TL megoldást, a CNN futása 41 percig, míg a TL 16 percig tartott.

5. ÖSSZEGZÉS

Az ipari termelésben a fogyasztói igények változatossága növekvő mértékben indokolja az agilis gyártási rendszerek használatát, amely a rugalmasság és a gyors reagálóképesség révén támogatja a kis szériás, egyedi termékek előállítását. A tanulmány egy mélytanulás-alapú hibafelismerési rendszert mutat be, amelyet kifejezetten agilis gyártási környezethez terveztünk, hogy valós idejű és költséghatékony minőség-ellenőrzést biztosítson. A rendszer fő célja a hagyományos minőségbiztosítási módszerek korlátainak áthidalása a különböző hibák azonosításában és osztályozásában, különösen az egyedi alkatrészek és a kis szériás termékek gyártásában.

A megközelítés egyik kulcseleme a szintetikus adatgenerálás, amely CAD modellek és a Blender szoftver segítségével hoz létre valóságghű referencia képeket különféle, gyakran előforduló hibákkal. Ezeket az adatokat egy konvolúciós neurális háló betanításához használjuk, amely így képessé válik a hibás alkatrészek valós idejű felismerésére a gyártóso-ron. A szintetikus adat használata jelentős előnyt biztosít az adatok annotációjának és előállításának költségei terén, különösen, mivel a ritkábban előforduló hibák modellezése és detektálása jelentős kihívást jelent a hagyományos adatalapú megközelítések számára.

A kifejlesztett mélytanulás-alapú hibafelismerési rendszer magas pontosságot és megbízhatóságot mutatott a hibák felismerésében, amelyet a kísérleti eredmények is alátámasztanak. Az eredmények alapján a rendszer sikeresen integrálható lehet a valós idejű minőség ellenőrzési folyamatokba, csökkentve az emberi beavatkozást

és javítva a gyártási folyamat általános hatékonyságát. Ez a megoldás költséghatékony és skálázható alternatívát kínál a hagyományos tesztelési és ellenőrzési módszerekkel szemben, különösen az agilis gyártási környezetek igényeihez igazodva.

A jövőbeni fejlesztések célja a rendszer valós gyártási környezetbe történő integrálása, ahol további finomhangolást és gyakorlati validációt lehet elérni. Emellett a megoldás alkalmazása más típusú hibák detektálására is kiterjeszthető lehet.

KÖSZÖNETNYILVÁNÍTÁS

A szerzők ezúton fejezik ki őszinte hálájukat a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatal által biztosított Egyetemi Kutatói Ösztöndíj Programnak, amelynek támogatása lehetővé tette a kutatás megvalósítását. Továbbá, köszönet illeti Monori János Bencét, aki értékes szakmai hozzájárulásaival segítette a kezdeti ötlet formálását, és közreműködött számos mélytanulási elem kidolgozásában.

HIVATKOZÁSOK

- [1] Yang Jing és tsai. „Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges”. *Materials* 13 (2020. dec.), 5755. old. DOI: [10.3390/ma13245755](https://doi.org/10.3390/ma13245755).
- [2] Yann LeCun, Yoshua Bengio és Geoffrey Hinton. „Deep learning”. *Nature* 521.7553 (2015), 436–444. old. DOI: [10.1038/nature14539](https://doi.org/10.1038/nature14539). URL: <https://doi.org/10.1038/nature14539>.
- [3] A Gunasekaran. „Agile manufacturing: A framework for research and development”. *International Journal of Production Economics* 62.1 (1999), 87–105. old. ISSN: 0925-5273. DOI: [https://doi.org/10.1016/S0925-5273\(98\)00222-9](https://doi.org/10.1016/S0925-5273(98)00222-9). URL: <https://www.sciencedirect.com/science/article/pii/S0925527398002229>.
- [4] Hugging Face. *Synthetic Data: Save Money, Time, and Carbon with Open Source*. Accessed: 2024-11-01. 2023. URL: <https://huggingface.co/blog/synthetic-data-save-costs>.
- [5] Alvaro Figueira és Bruno Vaz. „Survey on Synthetic Data Generation, Evaluation Methods and GANs”. *Mathematics* 10.15 (2022). ISSN: 2227-7390. DOI: [10.3390/math10152733](https://doi.org/10.3390/math10152733). URL: <https://www.mdpi.com/2227-7390/10/15/2733>.
- [6] MRPeasy. *Manufacturing Defects: Types, Causes, and Prevention Tips*. Accessed: 2024-11-10. 2023. URL: <https://www.mrpeasy.com/blog/manufacturing-defects/>.
- [7] Poly Haven. *Poly Haven - Free HDRIs, Textures, and 3D Models*. Accessed: 2024-11-07. URL: <https://polyhaven.com/hdris>.
- [8] TensorFlow Team. *Convolutional Neural Network (CNN) Tutorial*. <https://www.tensorflow.org/tutorials/images/cnn>. Accessed: 2024-11-07. 2024.
- [9] GeeksforGeeks. *ML - Stochastic Gradient Descent (SGD)*. Accessed: 2024-11-10. 2023. URL: <https://www.geeksforgeeks.org/ml-stochastic-gradient-descent-sgd/>.