Development of a Web Application with a Sheet Movement Visualization System for Tracking the Technological Process in a Sheet Rolling Shop

2025 International Russian Smart Industry Conference (SmartIndustryCon) | 979-8-3315-1124-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/SMARTINDUSTRYCON65166.2025.10985970

Rauza R. Abdulveleeva
Mathematics and Science dept.
Novotroitsky branch of the
"National Research Technological University MISIS"
Novotroisk, Russia
rashitovna-2011@mail.ru

Ildar R. Abdulveleev
Power supply of industrial enterprpises dept.
Nosov Mangnitogorsk State Technical University
Magnitogorsk, Russia
i.abdulveleev@magtu.ru

***Abstract*—The article describes the technology of developing a web application for a sheet rolling shop mnemonic diagram with a sheet metal movement visualization system and an algorithm for processing information flows about their location. The relevance of creating a mnemonic diagram is substantiated. The need to increase transparency in the sheet metal tracking and accounting system through visualization is shown, which is due to the problem of controllability of production processes that lead to downtime on units and corresponding violations in the steel sheet control and accounting system. The web application is a Django application developed using the Redis noSQL DBMS. With the help of the developed algorithm consisting of a sheet movement tracking module and a web interface, developers have the opportunity to debug the application without having to install equipment at the stage of designing and testing the web application. The information flow tracking algorithm, file structure and web page of the application, and the structure of JSON data transmitted to Redis are presented. This approach allows increasing the transparency and efficiency of the 2800 mill, increasing the speed of decision-making on managing production operations in the shop within a shift.**

Keywords—sheet metal tracking, web application, visualization, mnemonic scheme, JSON, Redis, Ajax

I. Introduction

In the modern metallurgical industry, where production processes are characterized by high speeds and large volumes, effective tracking and control of the movement of metal sheets in the sheet rolling shop is of critical importance. The ability to visualize and analyze sheet movement trajectories allows for increased productivity, reduced risks of loss and damage to materials, and timely and accurate data processing.

Russian metallurgical companies are actively implementing various technologies for tracking and visualizing the movement of metal sheets and other products in sheet rolling shops and at metallurgical enterprises [1-4]. For example, at the NLMK Lipetsk production site, a product tracking system based on laser scanners and global positioning systems (GPS) is used. Data on the movement of metal sheets and other products is visualized on interactive maps, which allows for optimization of logistics and increase the efficiency of production processes.

The Cherepovets Metallurgical Plant Severstal has implemented a project to implement a computer vision system to track the movement of steel slabs. The system uses cameras and image processing algorithms to determine the location of slabs in the workshop. Visualization of slab movement trajectories is carried out on interactive maps, which allows for optimization of logistics and production planning.

Visualisation plays a key role in modern industrial manufacturing, providing tools to better manage and optimise production processes. A challenge faced by many businesses is that production data often remains siloed, unstructured and unavailable for real-time analysis. This leads to a lack of transparency in processes, increased time and costs, as well as lower productivity and product quality.

In this context, visualisation becomes a hot topic, as it allows data on production processes to be presented in a visual form, which helps personnel better understand the current state of production and make prompt, more informed decisions [5-7].

The NLMK metallurgical plant (Russia) uses laser scanners to track the movement of products. The advantages of which are high accuracy, the ability to track indoors and outdoors. However, there are also limitations such as the high cost of equipment, the need for direct visibility for scanners, and GPS limitations in closed spaces.

At the Ural Steel metallurgical plant, they encountered a similar problem in the sheet rolling shop (SRM), where the existing tracking system in SRM 1 is based on lasers. However, in buffer zones, where many sheets are located in one place at once, lasers do not cope with their task. There is also a problem with heating the lenses of laser sensors, since under the mill where the lasers are installed there is a fairly high temperature, which melts the lenses of these lasers.

In this regard, the development of a web application with an integrated sheet movement visualization system is a pressing task aimed at increasing the efficiency and transparency of processes in the sheet rolling shop. Such a solution will allow real-time tracking of the movement of metal sheets, visualizing their trajectories, which will help personnel

979-8-3315-1124-1/25/$31.00 ©2025 IEEE

better understand the current state of production and provide analytical data for making prompt, more informed decisions [8-12]. In the light of this problem, a web application with a system for visualizing the movement of metal sheets and tracking their location on the 2800 mill was developed.

Application purpose:

- visual control over the situation in the rolling mill in terms of sheet movement in real time;
- saving and collecting information in the database on the movement of sheets, performing secondary operations with sheets;
- combining readings from video cameras, a modulator, with information on current sheet movements.

Problems to be solved:

- Reduction of unit failures due to sheet overlapping and loss of its position on the mill and in the accounting system;
- Increasing the transparency and efficiency of the workshop in the tracking system for produced sheets;
- Increased speed of decision-making on the management of production operations within the shift;
- monitoring the location of metal sheets on the 2800 mill at any time.

II. Description of the Application

The tracking system is designed to track the state of the sheet and the movement of metal sheets inside the workshop. For this, the workshop is divided into conditional sections, each of which has its own name and location coordinates. In each section, the position of the sheets can be tracked.

The mill can be divided into two parts: the part that rolls out the sheets and the part that processes the rolled sheets. It was decided to start tracking the sheets after the marking section, since the marking machine stores data on the current sheet that it marks. It was decided to read data on the sheets from it.

The problem being solved is the need to develop a web application and an algorithm for tracking information flows to determine the location of a metal sheet on a mill as a system capable of working with sheets that have both visible and invisible markings and ensuring the integrity of tracking their movement. The system is capable of determining changes in the movement of sheets by comparing data from cameras with previous iterations of the algorithm, and storing information in a database, sending it to the visualization system using the JSON interface [13-16]. The structure of the metal tracking system in the 2800 mill flow is shown in Figure 1 (Figure 1).

In the absence of the possibility to install the necessary equipment in the workshop, it was decided to develop a debugging ground - a mnemonic scheme of sheet movement. The web application is a mnemonic scheme developed using Django - a high-level Python web framework. Within the specifics of the Django framework, the application consists of two separate modules: a web interface and an image analysis system. The file structure of the application is shown in Figure 2.

The web interface includes the root folder ‘system2’, which consists of three other directories: ‘.venv’, ‘server’, ‘utils’; The virtual environment is located in the directory - ‘.venv’; ‘server’ is a directory that stores the ‘main.py’ file for running the program and two folders ‘static’ and ‘templates’. The ‘static’ and ‘templates’ folders store the ‘images’ directory with a set of images of sheets in png format. The ‘templates’ folder contains an html file that describes the appearance of the web interface, ‘utils’ is a directory with additional python utility files, needed to improve the work of interacting with the database, server and web interface.

The image analysis system is located in the root folder of ‘SheetTracking’ – the testing system. This folder includes 3 directories - data, photos, pic: ‘data’ – a folder storing JSON files with information about detected sheets, video fragments, descriptions of sheet sizes, shop parameters; ‘photos’ – a set of photo images created by the shop simulator program; ‘pic’ – a set of png images for creating a model of the rolling mill section.

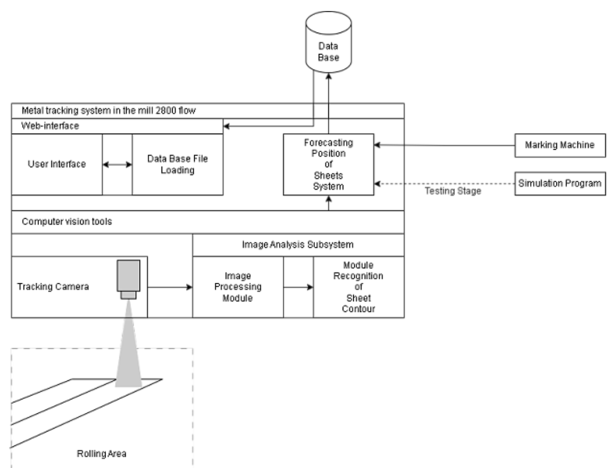

Fig. 1. Structure of the metal tracking system in the 2800 mill flow.

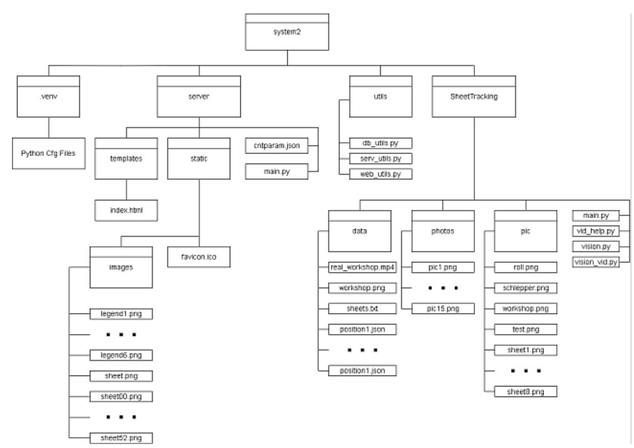

Fig. 2. File structure of a web application.

The root folder also includes python files: main.py – a demonstration stand; vid_help.py – a utility for working with video; vision.py – a program that detects sheets in a photo;

vision_vid.py – a program that determines the position of sheets in a video.

III. Operation of the Crane Movement Modulator

The tracking program consists of a data collection module, an analytical module, a database module, and a web module. An algorithm for processing the information flow has been developed for each module. The purpose and function of the data collection module is as follows: cameras are installed on parts of the mill in problem areas where a large number of sheets most often accumulate and they move non-linearly to obtain a video image file. The video file is processed using a program using the opencv library. This algorithm converts the video from the camera to a black and white format and looks for the darkest spots, which are the sheets. The processed video data is shown in Figure 3.

The coordinates of each spot, which is a sheet of metal, are determined by a previously created grid of the observed area. These coordinates are then written to a JSON file and transferred to the database. The format of the JSON file is shown in Figure 4.

The sheet recognition algorithm is supplemented by an analytical module that implements the sheet detection algorithm. This program receives JSON files from the data collection module and adds information for each sheet (slab ID, heat number, steel grade, order number). The algorithm receives this data from the marking machine in the form of a JSON file with a description of the data on the current sheet (Figure 5).

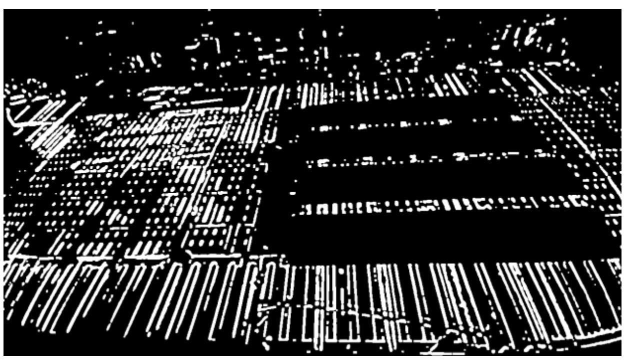

Fig. 3. Type of data received from the video camera.

```
[
    [sector X position, sector Y position, sheet X position, sheet Y position]...
]
```

Fig. 4. JSON file format with sheet position information.

```
{
    "id_slab": str
    "order": int
    "melting": str
    "batch": int
    "steel_grade": str
    "size": tuple[int, int, int]

}
```

Fig. 5. Structure of the transmitted JSON string.

The demonstration stand is a modulator, i.e. an additional program that replaces the work of the data module. It is modulated by the rolling mill section, to which the tester can add sheets. To add a sheet, you must first fill in the sheet data (Figure 6), then this data is generated in a JSON file and sent to the server using a POST request.

After pressing the "Send" button, the modulator will display the result of visualization of the position of the sheet metal on the mill to track its further movement. The result of the program's work with the sheet on the mill is shown in Figure 7. In this program window, you can start moving this sheet using the buttons ">", "V", "<" - to the right along the upper slapper, down along the roller table (the entire row moves), to the left along the lower slapper. With any movement of sheets, the program generates a new JSON file with the changed position of the sheets and sends it to the server.

This program is also responsible for handling errors in the tracking program. If sheets overlap, the data collection module identifies them as a single sheet and sends the coordinates of only one sheet. This error is processed using previous readings; if at some point in time a sheet disappears from the mill in a place where it could not have disappeared, the system accesses the database, receiving from it the previous version of the position in the form of a JSON file (Figure 2). Then it checks where each sheet could have ended up and adds the data to the JSON file storing the position of the sheets at this stage.

Sheets can only move in one direction at a time and cannot be removed from the flow at any point in the workshop, with the exception of perhaps one. Based on these properties, failures in the tracking system are excluded in situations where sheets move in an overlapping manner, or the camera fails to recognize a sheet. Figure 8 shows an example of an emergency situation where sheets are displayed in an overlapping manner and are identified as a single sheet.

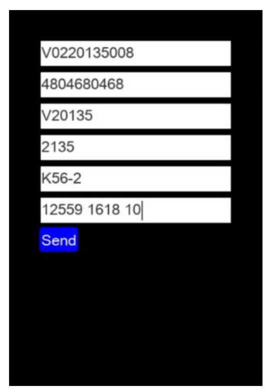

Fig. 6. Sheet data form.

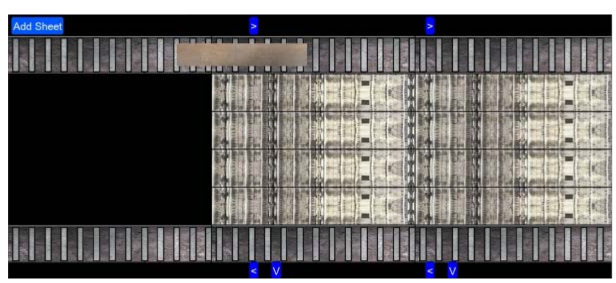

Fig. 7. Display of sheet positions on the mnemonic diagram.

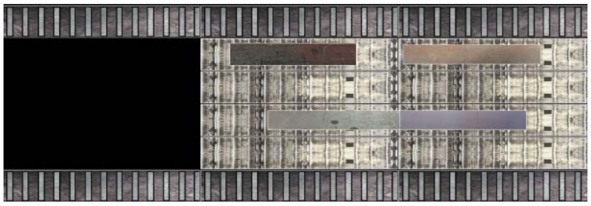

Fig. 8. An example of an emergency situation with overlapping sheets in a workshop area.

The condition that sheets can only move in one direction and cannot be removed from the flow except at one point allows to avoid system failures in case of complex sheet movements or if the camera is unable to recognize a sheet.

If the camera is unable to recognize a sheet, the algorithm predicts its location and waits for it to appear on the camera. If the sheet is not detected, the system requests information about its location from the user. If a sheet is lost at a point where it can be removed from the flow, the system requires the user to confirm or deny this action.

Depending on the area where the system is operating, sheets can be in a position with markings visible to the camera or invisible. It is proposed to ensure the integrity of the tracking system using a special algorithm. Figure 9 shows a general block diagram of the sheet location tracking algorithm.

The database stores all information about sheet movements and stores information in the program attributes associated with them. The database is based on the PostgreSQL DBMS and storing all information about sheet movements allows you to keep a history of events about the movement of metal sheets in the workshop. All information in the tracking system must be available for viewing by the user, for this purpose the website of the web application serves. The website displays information taking into account the requirements of a specific user.

For example, site operators are interested in operational information about the current state of the production process, as well as the ability to quickly respond to changes. It is advisable to develop an interface with one main window that displays the mnemonic diagram of the site, so as not to distract their attention from the entire mill. For other users, the entire mill is displayed. The site also has color coding, which is necessary to separate different heats.

In addition to color coding, it is possible to view information about each sheet in detail. In order not to overload the mnemonic diagram with excessive detail, which will make it difficult to perceive, it was decided to put this information in a separate window that opens when you click on a specific sheet.

To exchange information with the server, database and system modules, JSON files are mainly used. The primary source is a CCTV camera. It sends video to the system in mkv format, which is then processed by the image analysis system. The video identifies the locations of the sheets, and based on this data, two JSON files are generated: a file with data on the position of the sheets and a file with data on the new sheet (obtained from the marking machine) Sometimes there may be no information in one or both files at once. This is affected by the workload of the mill. In this case, these two files can be generated on a demonstration stand. This happens at the testing stage, when the use of the camera is not available.

Another option for obtaining information for the first file is the database, if it is not empty. This option also relates to testing and is designed to not generate new sheet locations on the demonstration stand, but to change the old positions of existing sheets.

After the two JSON files are generated, they are transferred to the sheet position prediction system. For stable operation of the algorithm, this system, using the presentation function for processing GET requests, sends a GET request to the server, and the server in turn returns a JSON file with the current sheet positions. The system compares the file from the server and the file received from the image processing module and, based on this data, generates a new JSON file with the sheet positions. Then, using an AJAX POST request, it sends the generated file to the server. The server, in turn, adds the new JSON file to the database [17-19].

The web page that displays the user interface with the sheet positions constantly sends GET requests to the server in order to receive changes in the sheet positions. The server sends the "newest" JSON file. Then the web page is rendered, and after successful rendering, the new image is displayed on the site.

In addition to updating sheet information, the web page may send a POST request to the server. This happens when the dispatcher makes manual changes to the sheet positions.

This function is added for emergency situations, if for some reason the system loses a sheet. Such changes are also generated as a JSON file and sent to the server using the view function for processing POST requests. This JSON file has the highest priority for the system and replaces the file with the current sheet position without any system checks.

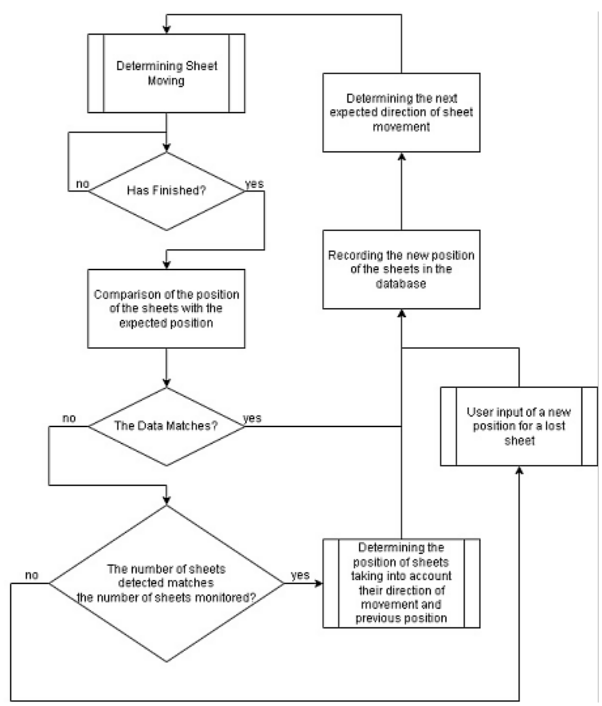

Fig. 9. Function of displaying cranes on the mnemonic diagram.

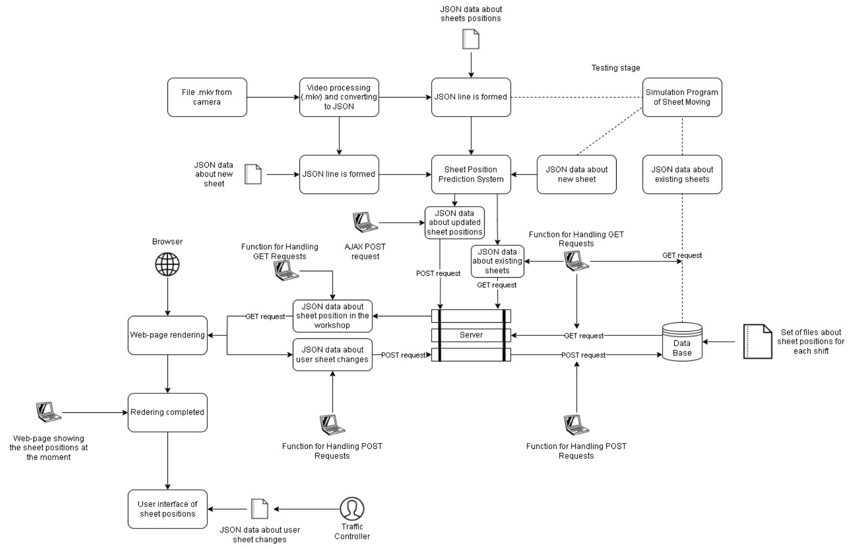

Fig. 10. Algorithm of the web application for tracking the movement of sheets of crane motion modulator operation.

Thus, the algorithm of the sheet movement tracking application in the form of an event-driven process (EPC) model is shown in Figure 10.

CONCLUSIONS

In the course of the work, a web application for a mnemonic diagram of sheet metal movement in a sheet rolling shop with a visualization and tracking system for sheet movement was developed. The proposed algorithm for modulating sheet movement makes it possible to debug the web application without the need to install equipment at the stage of designing and testing the web application. Implementation of the developed web application in a sheet rolling shop will increase the transparency and controllability of production processes, reduce material losses and optimize the use of resources. In addition, visualization of sheet movement will provide a deeper understanding of production flows and serve as a basis for making informed management decisions.

The developed mnemonic diagram with a visualization and tracking system for sheet movement is a promising solution to the problem of ensuring the integrity of the process of tracking their movement, due to the low transparency of the 2800 mill for the dispatcher and is taken for consideration for implementation in production by the management of Ural Steel.

REFERENCES

[1] I. I. Ivanov and P. P. Petrov, “Application of RFID technologies for tracking crane movements in metallurgical production,” Automation in Industry, no. 2, pp. 15–22, 2023.

[2] R. R. Abdulveleyeva and V. Korsakov, “Comparative analysis of methods and means of visualisation of cranes movement of electric steelmaking shop of JSC ‘Ural Steel’ for use in web application,” Digital Systems and Models: Theory and Practice of Design, Development and Application. Kazan, April 2024, pp. 270–274.

[3] A. A. Kuznetsov and P. P. Petrov, “Scalability of visualization systems in sheet rolling production,” Metals, no. 8(2), pp. 12–120, 2020.

[4] R. R. Abdulveleeva, V. A. Korsakov, and I. R. Abdulveleev, “Crane movement modulator for web application of electric steel melting shop mnemonic scheme with crane movement visualisation system,” 2024 International Ural Conference on Electrical Power Engineering (UralCon), Magnitogorsk, Russian Federation, 2024, pp. 812–816, doi: 10.1109/UralCon62137.2024.10718976.

[5] K. Healy, Data Visualization: A Practical Introduction. Princeton University Press, 2018.

[6] H. Shoppa, Technology of Steel Production Processes, Publisher, 2015.

[7] P. P. Petrov, “Development of a web-based visualization system for steel production using Django,” Industrial Informatics, vol. 14, no. 2, pp. 33–47, 2022.

[8] A. V. Ivanov, “Problems of implementation of computer vision systems at metallurgical enterprises,” Metallurgist, no. 5(3), pp. 22–28, 2019.

[9] D. Forsyth and J. Pons, Computer Vision. Modern Approach, Trans. from English. Moscow: Williams Publishing House, 2004.

[10] S. S. Sidorov, “Development of web applications using Django for visualization of production processes,” Software Products and Systems, vol. 35, no. 3, pp. 452–461, 2022.

[11] X. Li, “A Web-based visualization platform for industrial process monitoring using Django and WebSocket,” in IEEE Transactions on Industrial Informatics, vol. 17, no. 9, pp. 6379–6388, 2021.

[12] J. Wang, “An efficient data exchange method based on JSON for industrial Internet of Things,” in IEEE Access, vol. 7, pp. 82328–82341, 2019.

[13] D. D. Sokolov, “Application of JSON and AJAX for real-time data exchange in industrial automation web applications,” Modern High Technologies, no. 8, pp. 59–64, 2020.

[14] K. K. Kuznetsov, “Utilizing JSON and AJAX for real-time data exchange in industrial applications,” Information Technologies and Systems, vol. 11, no. 4, pp. 123–137, 2020.

[15] R. R. Abdulveleeva and V. Kravchenko, “Vulnerability analytics of web-application mnemonic scheme of electric steel-smelting shop with visualisation system of cranes and steel ladles movement,” Digital Systems and Models: Theory and Practice of Design, Development and Application, Kazan, April 2024, pp. 275–279.

[16] S. S. Smirnov, “Redis as a high-performance solution for real-time data processing in industrial applications,” Bulletin of Information Technologies, vol. 10, no. 2, pp. 88–102, 2019.

[17] H. Chen, “Redis-based real-time data processing architecture for industrial Internet of Things,” in IEEE Transactions on Industrial Informatics, vol. 16, no. 7, pp. 4769–4778, 2020.

[18] M. M. Volkov, “Redis as a high-performance solution for data processing in industrial visualization systems,” Scientific and Technical Journal of Information Technologies, Mechanics and Optics, vol. 19, no. 4, pp. 714–721, 2019.

[19] L.V. Radionova, A.D. Chernyshev, “Mathematical Description of AFE Rectifier Closed Loop System,” Procedia Engineering, vol. 129, pp. 16–21, 2015.