DOI 10.15276/imms.v12.no4.288
УДК 519.85.004.42

Informatics and Mathematical Methods in Simulation
Vol. 12 (2022), No. 4, pp. 288-295

A PROPOSAL OF STORAGE FOR CHROMOSOMAL DATA

O. Pysarchuk[1], Yu. Mironov[2]

[1]Igor Sikorsky Kyiv Polytechnic Institute. Peremogy Ave., 37, Kyiv, Ukraine, 3056, PlatinumPA2212@gmail.com
[2]National Aviation University, Guzar Lubomir Ave., Kyiv, Ukraine, 3056, 03058, yuriymironov96@gmail.com[2]

This article considers a problem of chromosomal pathologies detection. Chromosomal pathologies are dangerous and pose a great threat for family planning. To address them, the karyotyping process is conducted. Currently this process is manual or semi-manual, despite high effort and error cost. So, there is a need for automation of the process. This process (and automation algorithm) can be separated into different stages with various objectives. However, the shared element among these stages is format that allows to store and manage data efficiently. The goal of this paper is to propose such format. The paper revises peculiarities of karyotyping process and briefly describes steps of the pathology detection algorithm. It also considers common formats for bioinformatics data. However, their efficiency is debatable, since data stored in these formats is redundant for the task at hand. After that, a new custom data format is proposed. This format represents main entities involved in the process of anomaly recognition. Several fragments of algorithm are considered, and their complexity is estimated combined with proposed data format. As a result, this paper proposes a new data storage format used in a chromosome abnormalities recognition algorithm, and metrics that can be used to make measurable improvement over the proposed format.

Keywords: Algorithm complexity, data analysis, domain-driven design

Introduction

Congenital diseases have a significant impact on survivability rate of newborn children, including early deaths, miscarriages and stillbirth [1, 2]. Apart from this, further quality of life of surviving children may also be affected. A major subcategory of congenital diseases are chromosomal diseases [1].

These risks and issues are addressed by reproductive medicine. It introduces a wide variety of methods that handle various problems related to family planning and reproductive health. To solve the tasks of reproductive medicine, techniques of other branches of biology is used – for instance, cytogenetics. One of the main techniques for diagnosing chromosomal diseases is karyotyping [3].

Karyotyping is a process that implies an analysis of biological materials of parents/fetus, getting a visual depiction of their chromosomes and comparing chromosomes to ideograms – schematic depiction of "ideal" chromosomes [4]. Ideal chromosomes, called ideograms (fig. 1.A), are actually just a reference schematic representation used "by eye" to identify real chromosomes (fig. 1.B).

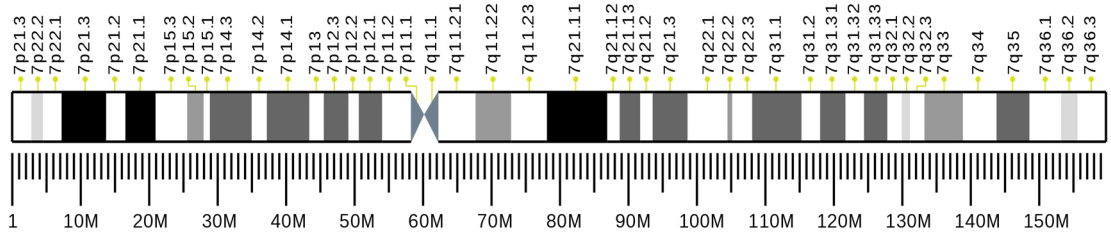

Fig. 1.A. Ideogram of human chromosome 7

Fig. 1.B. Pair of human chromosomes 7

While being time consuming and prone to human error, karyotyping is conducted manually, with limited means of automation. There are some solutions that present partial automations (like Lucia Karyo [5]), but they mostly provide useful utilities than true means of automation.

Proposing a system that offers automation of chromosomal recognition is a large undertaking that can be divided into multiple separate units of research in various fields: Computer Vision, algorithms, decision making systems [6]. Full automation of chromosomal recognition includes:

- Extracting data from visual depiction of live chromosomes;
- Extracting data from ideograms;
- Compare "parsed" chromosomes and ideograms with respect to certain rules;
- Make a decision – a basis for medical diagnosis;

In order to perform all these actions efficiently, it is crucial to store and manipulate data in an efficient manner. Therefore, it is necessary to design a data format that will complement the algorithm.

Related Papers

The general problem of programmatic chromosome pathologies recognition is not entirely new – there are even ready commercial solutions for reproductive laboratories to use. However, the common issue is that they do not create a comprehensive pipeline that would be able to get input image and produce a diagnosis. The aforementioned Lucia Karyo [5] can serve as an example – while it is helpful and effective for improving efficiency of karyotyping, it mostly consists of a database and specific image editing features. Moreover, no open data formats designed specifically for chromosomes have been found.

The global task under consideration is related to the field of bioinformatics, and there are several well-known formats for storing biological data. The prominent examples are Variant Call Format [7] and Stockholm Format [8], but there is also a considerable variety of accepted ways to store bioinformatic data [9].

Variant Call Format is a format that stores information about positions in the genome. One of its main features is brevity – it only stores differences from the norm. VCF focuses on the unit of measurement that is not applicable to the problem considered by this article, but the idea of storing only differences might potentially grant a boost to storage format and algorithm efficiency. It should also be noted that VCF operates a flat array of data, while chromosome pathology might need some data nesting to represent itself in full.

Stockholm format, like VCF, operates genome sequence alignment. However, it has vastly different syntax and shares richer metadata features. Nonetheless, using Stockholm format for the purpose of this article seems as redundant as using VCF.

However, these formats are mostly centered around genetic information, which is redundantly precise for the task at hand. Therefore, there might be a need to develop custom format centered around chromosomes instead of genes.

Goal

The goal of the paper is to propose a data storage format designed for efficient storage of chromosomal data, that would be fit to handle both recognized “live” chromosomal data and ideogram data.

Such a format would be useful for algorithms focused on chromosome management.

Main Body

In order to formulate a proposal of a data format, it is necessary to revise the process of karyotyping domain and the general algorithm of its automation.

The goal of karyotyping process is to gather information about chromosomes of a person and compare them to the “reference” versions, known to be normal and healthy. By comparing actual and reference chromosome, mismatches can be found and diagnosis can be made. The process can be roughly separated into three stages.

The first stage of karyotyping process includes gathering of the material and preprocessing. This process is out of scope of the research since it it heavily relies on specific medical hardwar. This stage results in a metaphase plate – a random arrangement of chromosomes depicted on fig. 2.

Metaphase plate includes all the chromosomes of a person, plus possible noise and obstacles.

Fig. 2. Metaphase plate

The second stage implies recognition of metaphase plate contents. Metaphase plate contains all the chromosomal data of a person, but it is not arranged and insufficient to state a diagnosis. In this stage, a medical specialist has to manually inspect a metaphase plate image, remove noises, obstacles and categorize each chromosome.

A healthy karyotype consists of 22 “numbered” pairs and XX/XY pair for female/male patients respectively. Therefore, there are 24 types of chromosomes – 22

"numbered" ones, X and Y. Each of these 24 "types" has a distinctive visual pattern, consisting of specific combination of black and white stripes (bands) of specific lengths. So, in order to categorize a chromosome, specialist has to compare it to a schematic representation of an ideal chromosome called "ideogram" (fig. 1.A)

Categorized chromosomes are arranged in pairs, forming a karyogram – an intermediate result of a karyotype (fig. 3).

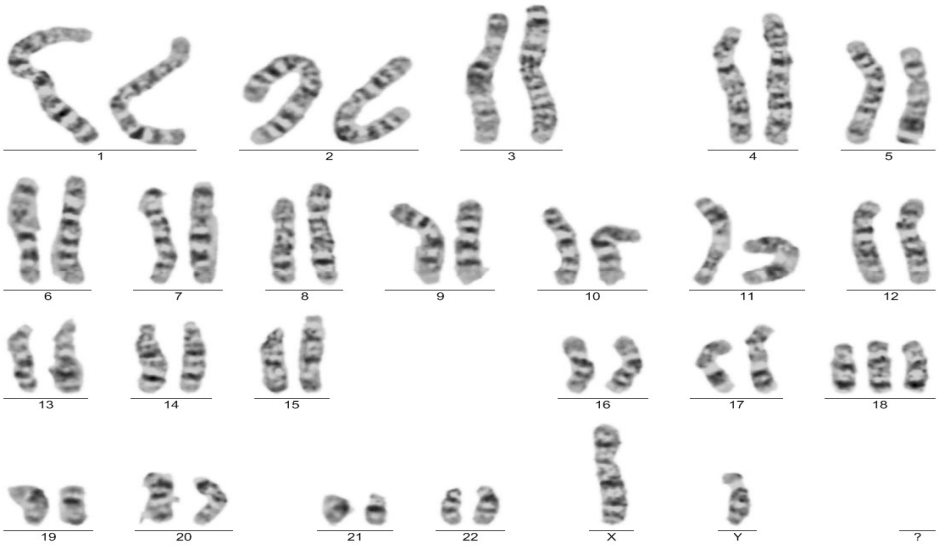

Fig.3. Karyogram with trisomy 18

Karyogram is a reference image for more convenient visual analysis, allowing to detect anomalies. There are several major groups of anomalies to detect:

- Quantitative anomalies. Chromosomes commonly go in groups of two, but sometimes there are three (trisomy) or more chromosomes of the same type. Notorious examples are Down syndrome (trisomy 21) [10], Edwards syndrome (trisomy 18) [11], but there are more syndromes caused by redundant number of chromosomes. Detecting a bigger-than-expected number of healthy chromosomes is a sign of quantitative anomaly;
- Structural anomalies. A malformed chromosome is something that is called a structural anomaly. Common examples of structural anomalies are duplication (same part of a chromosome repeating twice), deletion (part of a chromosome is missing) and translocation (part of a chromosome is moved) [12]. Detecting a partial match of a chromosome may help detecting a structural anomaly, but many checks are required to confirm the exact nature of such anomaly;

The third stare results in the actual diagnosis. Medical specialist analyses a combination of detected anomalies and states the final result.

Having overviewed the process of karyotyping, it is possible to revise a general algorithm for its partial automation. The algorithm can be broken into separate atomic parts which could be addressed separately. Each part addresses specific problem, can be improved independently or mocked. When combined, these parts form a pipeline that takes chromosome image as input and a suggested diagnosis as an output.

This paper is focused on a data format used in such an algorithm, but brief algorithm overview is necessary for the context.

The first stage remains unautomated, since it depends on advanced medical hardware. The metaphase plate (fig. 2) is considered to be the input data for the algorithm.

The second stage starts from an image and its goal is to recognize its content. As an input image, anything depicting a set of chromosomes can be used, such as metaphase plates and karyograms. In realistic scenarios, metaphase plates would be used, but user could also upload a karyogram to validate manually conducted karyotyping.

In any case, this stage starts from noise removal from the image (which is basically anything that does not have features of a chromosome). After that, a contour detection is conducted, identifying tangible objects inside an image. The detected contours are filtered again, removing objects that do not resemble chromosomes by a set of criteria. This way, algorithm would be able to continue working with a definitive set of visual objects.

The next step of this stage is extracting features from each specific visual object. This step is iterative and considers one object at a time, attempting to detect information that could later be used to determine

During this step, the algorithm loops through the set of objects and extracts features from each of them, moving data into a custom format considered by this article. Having extracted image data into this format, it is possible to compare parsed chromosomes to ideograms (also described into same custom format).

This comparison, as well as identifying chromosomal diseases, constitute the third step. This would require a separate decision-making mechanism [13], that would also account for variable chromosome sizes and would detect chromosome being a mutated version of a certain ideogram. And, having a set of detected chromosomes and their abnormalities, it is possible to determine a suggested diagnosis. In order to map abnormalities to a distinct diagnosis, a knowledge base is required. So, it would be logical to keep a "dictionary" with possible chromosome states and corresponsive diagnoses. It should be noted, that not each chromosome abnormality has a distinctive name assigned to it – but it can be surely stated that if there is any mismatch with healthy karyotype, an anomaly takes place.

Since algorithm is focused on chromosome storage and comparison, designing efficient data format is essential for its effectiveness. This format should be able to represent a set of chromosomes, where each chromosome consists of bands with specific color and size. Fig. 4 shows a class diagram with suggested data format.

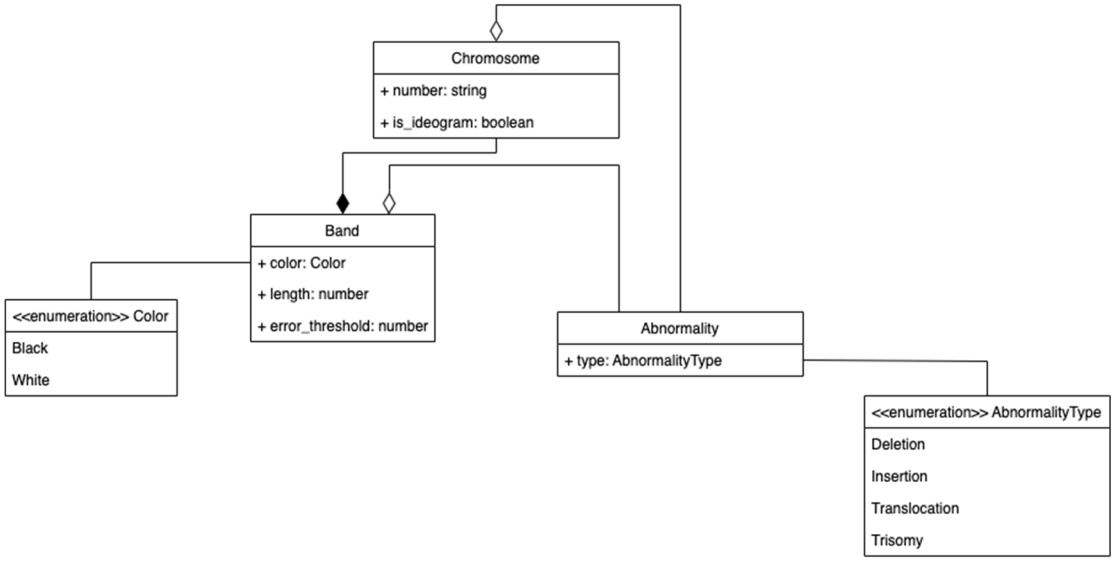

Fig. 4. Class diagram depicting proposed data storage

The *Chromosome* is the main object of the storage. It stores information of a single chromosome. The *number* property stores a suggested number of chromosome – a value from range 1…24, X or Y. *is_ideogram* flag shows what data is actually stored in this instance – a real chromosome or ideogram. This way all the data used in algorithm is kept in a uniform format.

Each chromosome has nested list of *Bands* – a colored segments of chromosome. Each segment has *color* and *length.* Bands could be a result of feature detection from image, or just a serialization of ideogram for more straightforward comparison. Since chromosomes could be of a variable size, it is proposed to store the relative length. This

way it would be possible to compare chromosomes by their internal features despite size differences.

There is a hypothetical issue comes from a fuzzy nature of data – it is improbable that there would be exact match between ideal band and a patient band. To address this issue, *error_threshold* field is added to a band. It stores maximum allowed deviation between ideal bend and chromosome bend.

The aforementioned objects are created as a part of chromosome feature extraction. After that, parsed chromosomes are compared against ideograms, bend by bend, to find a perfect match. On this stage, quantitative anomalies could be detected – this happens when there is an improper number of valid chromosomes. All the cromosomes that do not have a perfect match along with ideograms, are compared to ideograms bend by bend to find the closest match. When found, a structural abnormality is registered – this means that a part of chromosome is missing or duplicated.

Anomalies are stored into the *Abnormality* entity. This entity has a *type* and references to chromosomes and bends. In case of quantitative anomaly, reference to a chromosome is sufficient. In case of structural anomaly, affected bands are referenced.

In a stage of diagnosis making, a list of abnormalities is traversed and compared to a list of known medical conditions. If there is a match – a diagnosis is proposed. If there are no matches, but there are still abnormalities – it means the patient has an unidentified pathology, which is not uncommon. If there is no supposed diagnosis and there are no abnormalities – the patient's chromosomes are considered healthy.

Having described the nature of data format, it is necessary to evaluate its efficiency of executing core algorithm operations on proposed data format. Suppose there is a set of N chromosomes and a set of M ideograms. Each chromosome n consists of B_n bands, and ideogram m consists of B_m bands. In this case:

Table 1

Operations complexity

	Operation	Complexity
1.	Identification of a single chromosome. A chromosome is iteratively compared to each ideogram *m.*	*O(M)*
2.	Identification of quantitative abnormality. Each chromosome *n* is compared to each ideogram *m.* After that, each detected chromosome is looped to find unwanted duplicates.	*O(N*M+N)*
3.	Identification of a deletion. Each chromosome *n* is compared to each ideogram *m*. If a match is detected and it is partial, it is supposedly a deletion.	*O(N*M)*

The provided tests are fairly simple, but they should be sufficient to demonstrate the nature of data, serve as a reference for further research and efficiency measurement. Moreover, lack of open formats designed to store chromosomes, it is important to establish a baseline for improvements and comparison.

Results and Discussion

Unfortunately, not much could be reused from the research of existing storage formats of biological data. Formats like VCF or Stockholm formats are too detailed and store too much information about chromosomes to be efficient for the task at hand.

Due to the lack of research dedicated to designing data storage of chromosomes, it is hard to state that the proposed storage format offers outstanding efficiency. However, it represents application data domain in a straightforward and non-redundant way, introducing code entities matching domain objects. This would allow to reliably represent complexity of operations that are performed during cytogenetic analysis, and would allow to introduce measurable algorithms for its automation.

Conclusions

Automation of chromosomal diseases recognition is a relevant problem. Currently it is done in a manual or semi-manual way – and due to its high effort consumption and high cost of error it requires automation. This paper proposes a data format used for storing data for automation algorithm.

The essence of the problem has been revised, describing the nature of karyotyping process as well as general automation algorithm. Existing data formats used in domain of biocybernetics are revised, and none of them proved to be efficient for the task at hand.

After considering the requirements that should be addressed, a new custom data format has been suggested. This format stores both chromosomes and ideograms as a collection of bands, allowing easier comparison because of similar nature of data. It also stores detected abnormalities in separate objects, mostly for caching reasons.

A simple set of operations has been described and its potential complexity has been evaluated. This would allow to make measurable improvements in further research, establishing a metrics baseline.

References

1. Wapner RJ. Genetics of stillbirth. *Clinical Obstetrics and Gynecology*. 2010. Vol. 53(3), P.628-34. URL: https://doi.org/10.1097/GRF.0b013e3181ee2793
2. Hay S.B. ACOG and SMFM guidelines for prenatal diagnosis: Is karyotyping really sufficient? *Prenatal Diagnostics*. 2018. Vol. 38(3). P.184-189. URL: https://doi.org/10.1002/pd.5212
3. O'Connor. Karyotyping for chromosomal abnormalities. *Nature Education.* 2008. URL: https://www.nature.com/scitable/topicpage/karyotyping-for-chromosomal-abnormalities-298/
4. Cytogenetics L., Karyo L. URL: https://www.lucia.cz/en/products/lucia_karyo
5. Pysarchuk O., Mironov Y. A Proposal of Algorithm for Automated Chromosomal Abnormality Detection. *Modeling, Control and Information Technologies: Proceedings of International Scientific and Practical Conference*. 2021. Vol. 5, P.83–86. URL: https://doi.org/10.31713/MCIT.2021.26
6. Samtools team. The Variant Call Format Specification. 2022. URL: https://samtools.github.io/hts-specs/VCFv4.3.pdf;
7. Sonnhammers E. Stockholm format. URL: https://sonnhammer.sbc.su.se/Stockholm.html
8. Wikipedia, Wikimedia Foundation. Biological sequence formats. URL: https://en.wikipedia.org/wiki/Category:Biological_sequence_format
9. Antonarakis S.E. Down syndrome. *Nature Reviews Disease Primers*. 2020. Vol.6(1). P:9. URL: https://doi.org/10.1038/s41572-019-0143-7
10. Outtaleb F.Z. Trisomy 18 or postnatal Edward´s syndrome: descriptive study conducted at the University Hospital Center of Casablanca and literature review. *Panafrican Medical Journal*. 2020. Vol.37, P.309. URL: https://doi.org/10.11604/pamj.2020.37.309.26205;
11. Clancy S., Shaw K. DNA deletion and duplication and the associated genetic disorders. *Nature Education*. 2008. Vol. 1(1). P.23. URL: https://www.nature.com/scitable/topicpage/dna-deletion-and-duplication-and-the-associated-331/
12. Pysarchuk O., Mironov Y. Chromosome Feature Extraction and Ideogram-Powered Chromosome Categorization. *Advances in Computer Science for Engineering and Education. ICCSEEA 2022. Lecture Notes on Data Engineering and Communications Technologies*. Springer, Cham. 2022. URL: https://doi.org/10.1007/978-3-031-04812-8_36;

O. Pysarchuk, Yu. Mironov

ФОРМАТ ЗБЕРІГАННЯ ХРОМОСОМНИХ ДАНИХ

О.О. Писарчук[1], Ю.Г. Міронов[2]

[1] Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». 03056, м. Київ, Солом'янський район, пр-т Перемоги, 37. , PlatinumPA2212@gmail.com
[2]Національний авіаційний університет, 03058, Київ, пр. Гузара Любомира, yuriymironov96@gmail.com

Публікація розглядає задачу автоматизованого розпізнання хромосомних патологій. Хромосомні патології представляють значну небезпеку при плануванні вагітності. Для протидії цій проблемі задіюється процес каріотипування. На даний момент такий процес проводиться вручну чи в напівавтоматичному режимі, не дивлячись на великі часові затрати та високу ціну помилки. Отже, існує потреба в автоматизації даного процесу. Процес каріотипування (як і алгоритм його автоматизації) можна поділити на кілька етапів з різними цілями. Але у цих етапів є спільний елемент – формат, в якому зберігаються та обробляються дані. Метою даної статті є пошук такого формату. В статі оглянуто особливості процесу каріотипування та коротко описані кроки алгоритму з розпізнавання хромосомних патологій. Також розглянуто поширені в сфері біоінформатики формати даних. Нажаль, їхня ефективність у застосуванні до представленої задачі є сумнівною. Публікація пропонує новий формат даних для вирішення проблеми. В форматі представлені основні сутності задіяні в процесі розпізнавання аномалій. Розглянуто кілька фрагментів алгоритму, і розглянута їхня ефективність в комбінації з форматом даних. Результатом даної статі є запропонований формат для зберігання та обробки даних, що може бути використаний в процесі автоматичного розпізнавання хромосомних аномалій. Отримані заміри можна використовувати для подальшого покращення результативності алгоритму та ефективності формату зберігання даних.

Ключові слова: складність алгоритму, аналіз даних, domain-driven design