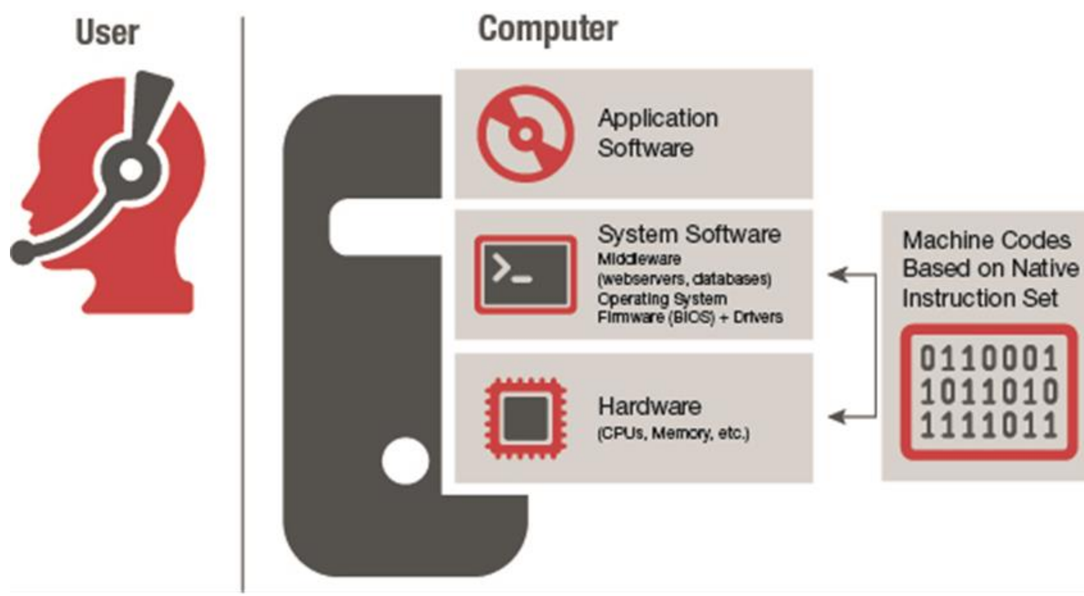
BakerHostetler



Patent Drafting for Machine Learning: Structural Claim Limitations, Avoiding Sections 101 or 112 Rejections, Drafting for EPO

Michael D. Stein,

BakerHostetler LLP

Presented for WSPLA, September 18, 2019

Outline

BakerHostetler

1. Technology Background
2. §112 Issues Affecting AI/ML Inventions
3. USPTO Guidance on Patent Eligibility (2019)
4. Patenting AI at the EPO

Technology Background

Artificial Intelligence

+

Machine Learning

What is Artificial Intelligence?

- Artificial intelligence (AI) is the ability of a computer program or a machine to think and learn. It is also a field of study which tries to make computers "smart."
- Example use cases include: understanding human speech, playing strategic games (e.g., chess), self-driving cars, and interpreting complex data.
- An extreme goal of AI is to create computers that can learn, solve problems, and think logically.
- AI involves a combination of many different fields, such as computer science, mathematics, linguistics, psychology, neuroscience, and philosophy.

Long-Term Problems in AI

BakerHostetler

- Creating a “general artificial intelligence,” which can solve many different problems, instead of focusing on just one.
  - Including making new discoveries or inventions???
- Developing a machine capable of “perceiving its environment” and making decisions or taking actions accordingly, like a human.

AI Sub-Field

Search and Optimization

- Intelligently searching through many possible solutions, taking advantage of machine’s high processing speed.

AI Sub-Field

BakerHostetler

Neural Networks

- Neural networks are modeled after the neurons in the human brain, where a trained algorithm determines an output response for input signals. Multiple layers process an input, with each layer applying more and more complex processing.

AI Sub-Field

Control Theory

- Control of continuously operating dynamical systems in engineered processes and machines. The objective is to develop a control model for controlling such systems using a control action in an optimum manner without delay or overshoot and ensuring control stability.

What is Machine Learning?

- Machine Learning gives “computers the ability to learn without being explicitly programmed.”
- Machine learning explores the study and construction of algorithms that can learn from and make predictions on data – such algorithms overcome following strictly static program instructions by making data-driven predictions or decisions, through building a model from sample inputs.
- Example applications include email filtering, detection of network intruders, optical character recognition (OCR), and computer vision.
- Source: *Wikipedia: Machine Learning* [https://en.wikipedia.org/wiki/Machine_learning]

Types of Machine Learning

- Supervised Learning: The computer is presented with example inputs and their desired outputs, given by a “teacher,” and the goal is to learn a general rule that maps inputs to outputs.
  - E.g. The animals in the picture are labeled as “cat” or “dog”, and the computer trains itself to identify other animals, in other pictures, as either “cat” or “dog”.

Types of Machine Learning

BakerHostetler

- Unsupervised Learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
  - E.g. Classify the animals below into different types.

Types of Machine Learning

- Reinforcement Learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). The program is provided feedback in terms of rewards and punishments as it navigates its problem space.
  - E.g. Learn to drive…

Example ML Claim Ver. A

1. A method comprising:

accessing, at one or more computing devices, an image;

determining whether the accessed image includes a cat; and

providing an output indicating whether the accessed image includes the cat.

Example ML Claim Ver. B

BakerHostetler

1. A method comprising:

accessing, at one or more computing devices, an image;

determining, using an image classification engine stored in one or more memories of the one or more computing devices, whether the accessed image includes a cat, the image classification engine being trained, using unsupervised learning, to classify images into a plurality of classes and being notified, via a graphical user interface (GUI), which classes correspond to cats, the GUI displaying at least one image from each class of the plurality of classes; and

providing an output indicating whether the accessed image includes the cat.

Example ML Claim Ver. C

BakerHostetler

1. A method comprising:

accessing, at one or more computing devices, an image;

determining<u>, based on a feature vector comprising a plurality of features that are extracted from the accessed image,</u> whether the accessed image includes a cat<u>, the plurality of features comprising at least an identification of a portion of the accessed image as representing a paw, a symmetry of the paw, and a shape of the paw</u>; and

providing an output indicating whether the accessed image includes the cat.

§112 Issues Affecting AI/ML Inventions

I. Problem Overview

BakerHostetler

Functional Claims

Can cause:
- **Overbreadth**
- **Unclear Scope**
- **“Preemption”**

Undue preemption can:
- stifle competition
- tax innovation
- create wasteful litigation

Litigation

<u>Legal Solutions</u>:
- *Use § 101 to invalidate overly broad (“abstract”) claims (Alice/Mayo)*
- *Use § 112(b) to invalidate unclear claims (Aristocrat)*

<u>Problem</u>:
- *Courts lack guidance/procedure to effectively apply the law.*
- *Courts **lack “context”** for evaluating claims from perspective of POSITA.*
- ***Too many patents are being invalidated on incorrect grounds!***

USPTO

<u>Alternative Solution</u>:
- *Include **structural limitations** in claims for software-related inventions.*
- *Applicants and USPTO ensure patent file history reflects examiner’s consideration of § 101 and § 112 issues.*
- *Provide “context” for courts.*

Background

BakerHostetler

Critics argue:

- Patents do less good and cause more harm in the software industry than in other industries.
- The aggregation of overbroad and unclear claims can result in innovation-stifling patent thickets.
- Software patents are not necessary to spur innovation because software innovation is less costly than innovation in the life sciences.

Why functional claim limitations are used in software patents

BakerHostetler

Why functional claiming is so common in software patents:

1. Software structure and function can be separated. A software developer can write new software without knowing details of the hardware.
2. The software industry lacks a commonly accepted “vocabulary” for defining software elements. In software, a broad claim requires defining the invention at a higher level of abstraction. Software developers coin new terms to define the functional elements of the software, and the meaning and scope of such new terms is often not explicitly defined.

Functional claim limitations and 35 USC §112(f)

BakerHostetler

35 U.S.C. 112 Specification.

. . .

(f) ELEMENT IN CLAIM FOR A COMBINATION.—An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

Aristocrat Techs. Australia Pty Ltd, v. Int'l Game Tech., 521 F.3d 1328 (Fed. Cir. 2008)

Claim

Claim 1 reads as follows:

A gaming machine

having display means arranged to display a plurality of symbc
display format having an array of n rows and m columns of
positions,

game control means arranged to control images displayed on the
means,

the game control means being arranged to pay a prize v
predetermined combination of symbols is displayed
predetermined arrangement of symbol positions selecte
player, playing a game, including one and only one symbol
in each column of the array,

the gaming machine being characterised in that selection mea
provided to enable the player to control a definition of one
predetermined arrangements by selecting one or more
symbol positions and

the control means defining a set of predetermined arrangement
current game comprising each possible combination of the
positions selected by the player which have one and o
symbol position in each column of the display means,

wherein the number of said predetermined arrangements for a
game is a value which is the product k_1 . . . x . . . k_i . . . x
where k_i is a number of symbol positions which have been s
by the player in an i^{th} column of the n rows by m columns of
positions on the display ($0 < i \leq m$ and $k_i \leq n$).

Disclosed Structure

In making this argument, Aristocrat relies on Figure 1 and Table 1
patent, which provide examples of how player selections translate to possible
combinations:

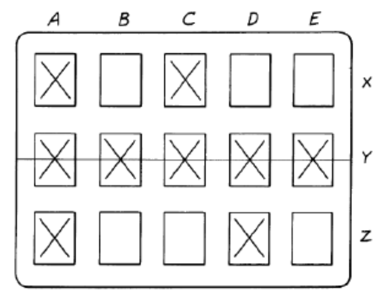


FIG. 1

TABLE 1

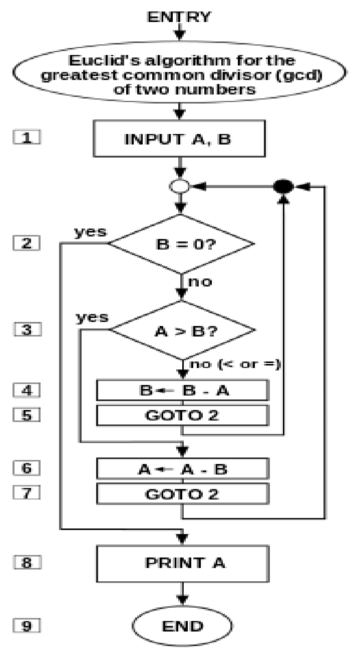
LINE NO	DISPLAY POSITIONS USED			
1	AX	BY	CX	DY
2	AX	BY	CX	DZ
3	AX	BY	CY	DY
4	AX	BY	CY	DZ

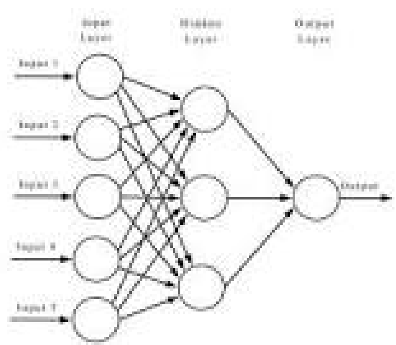
TABLE 1-continued

LINE NO	DISPLAY POSITIONS USEI			
5	AY	BY	CX	DY
6	AY	BY	CX	DZ
7	AY	BY	CY	DY
8	AY	BY	CY	DZ
9	AZ	BY	CX	DY
10	AZ	BY	CX	DZ
11	AZ	BY	CY	DY
12	AZ	BY	CY	DZ

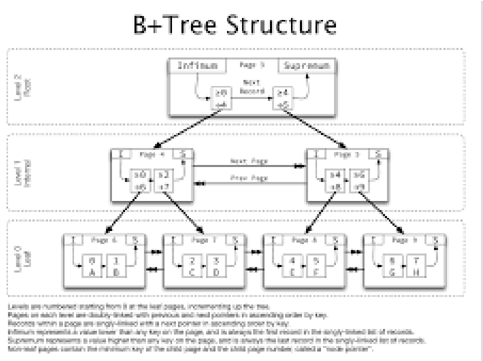
Structural claim limitations for software

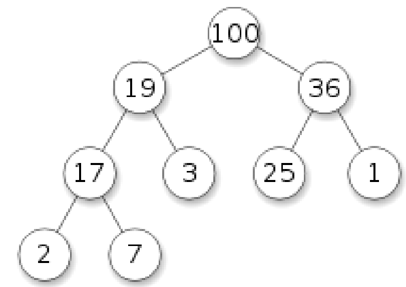
Program structures

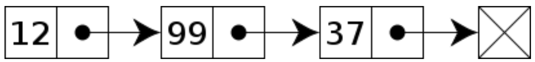




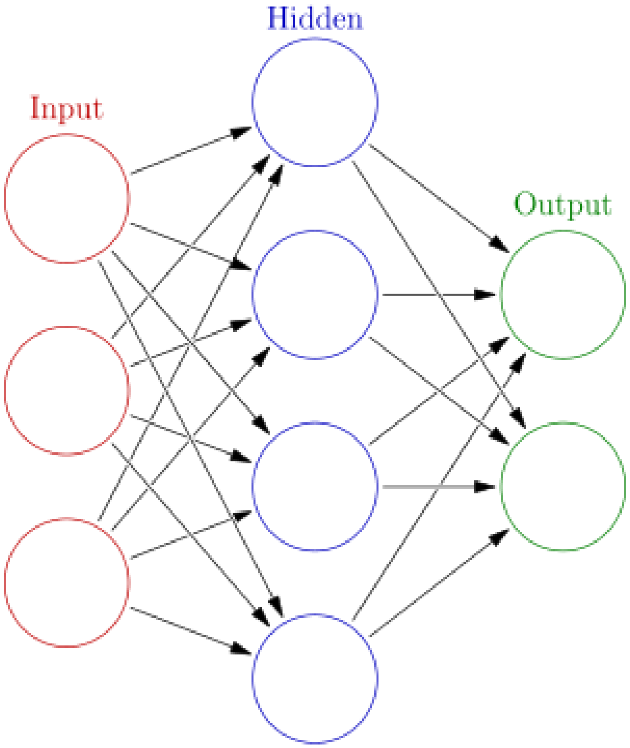
Data structures

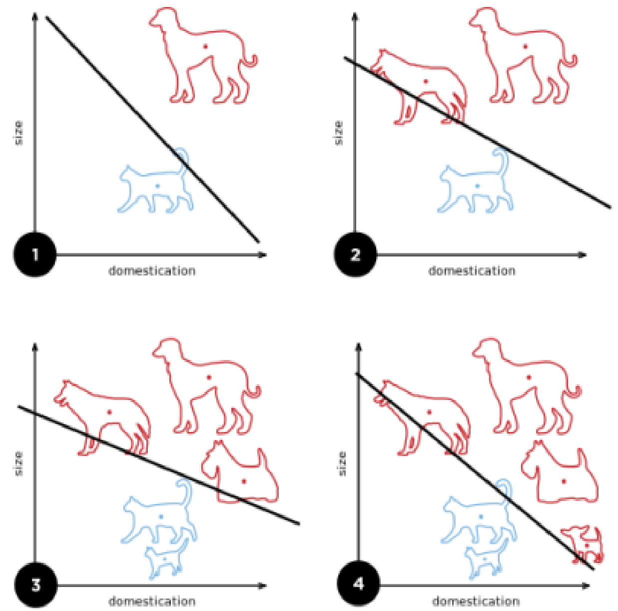






Example: Artificial Neural Network





Example claim to computer-implemented system comprising an ANN

BakerHostetler

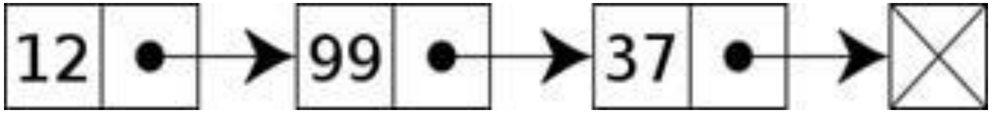
1. A classification system, comprising:

a **processor** configured to execute instructions programmed using a predefined set of **machine codes**; and

an **artificial neural network** (ANN) comprising:

> first, second, and third **input nodes**, wherein each input node includes a memory location for storing an input value;
>
> first, second, third, and fourth **hidden nodes**, wherein each hidden node is connected to each input node and includes computational instructions, implemented in machine codes of the processor, for computing first, second, third, and fourth output numbers, respectively; and
>
> first and second **output nodes**, wherein the first output node includes a memory location for storing a first output signal indicative of a first classification, and the second output node includes a memory location for storing a second output signal indicative of a second classification.

Example: Linked List Data Structure and Linear Search Algorithm



Boolean logic expressions and operators: Logical operands and logical operators are combined together. Operands are statements (that can be proven True or False) and the operators are logical AND, OR and NOT. Boolean expressions involve comparison operators that can be evaluated to determine if they are True or False. Comparison operators include: $= < > \geq \leq \neq$.

Expressions combining Boolean operators and comparison operators are written and evaluated in computer programming languages using control statements that control which sections of code in a program are executed. **Decision or Conditional statements** are a type of Control statement and are often referred to as **IF..THEN..ELSE** statements.

Iterative constructs, or Loops, allow a section of code to be repeated. There are several variations to iterative or looping constructs but for the most part they fall into two categories**: FOR loops and WHILE/DO loops**.

Pseudocode for a linear search algorithm implemented with Boolean logic and FOR and IF..THEN loops:

```
procedure linear_search (list, value)
  for each item in the list
    if match item == value
      return the item's location
    end if
  end for
end procedure
```

Example claim to computer-implemented search system comprising a linked list data structure

1. A search system, comprising:

a **processor** configured to execute instructions programmed using a predefined set of **machine codes**;

memory coupled to the processor;

a **linked-list data structure** implemented in said memory, comprising a **first node**, a plurality of **intermediate nodes**, and a t**erminator**;

wherein the first node is connected to a first intermediate node;

wherein each of intermediate nodes except for a last intermediate node is connected to a subsequent one of the intermediate nodes; and

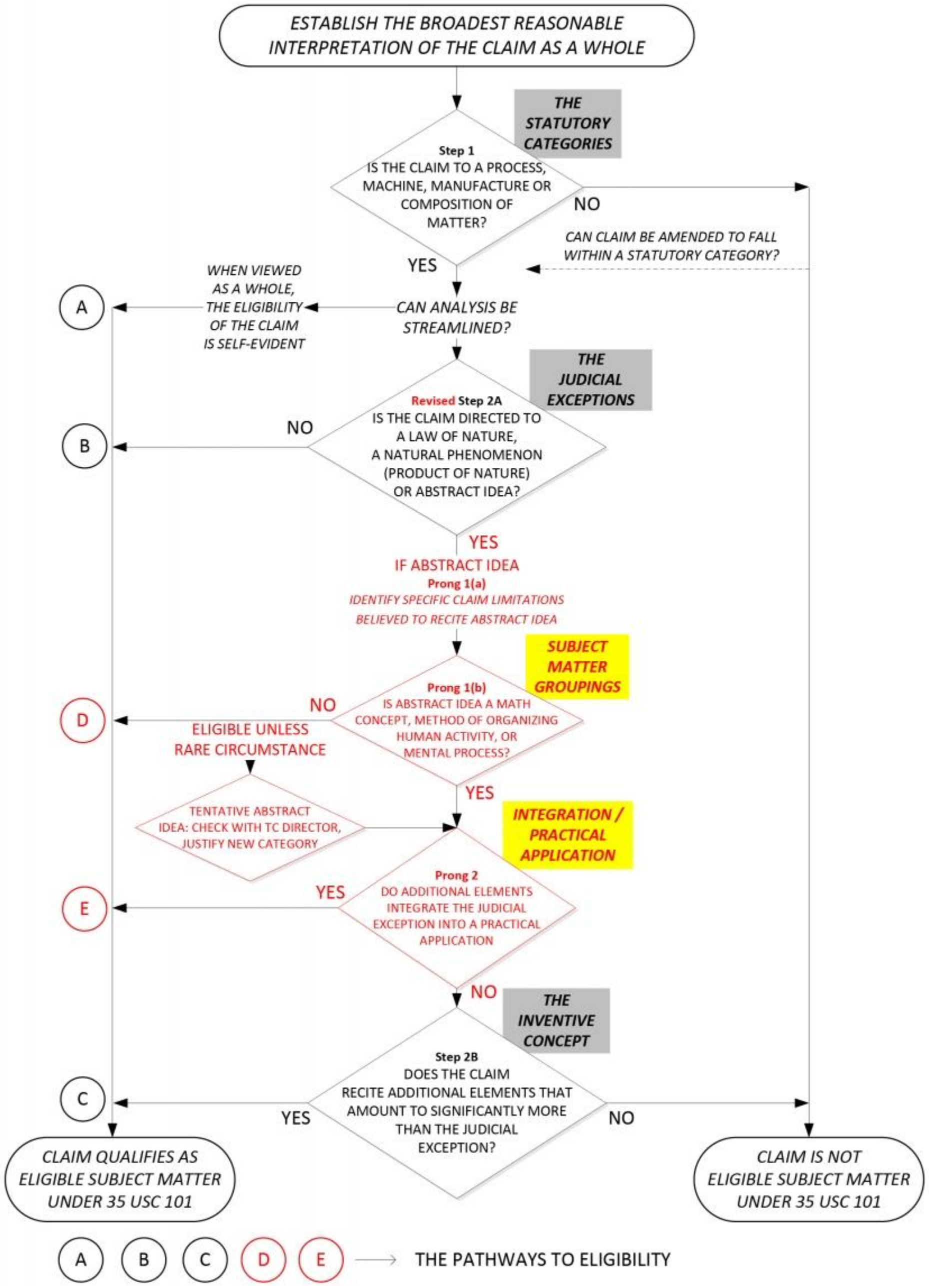
wherein the last intermediate node is connected to the terminator; and

a **linear search algorithm** implemented in machine codes for the processor.

2. The system of claim 1, wherein the linear search algorithm comprises an **iterative construct** and a **Boolean logic** expression.

USPTO: 2019 Revised Patent Subject Matter Eligibility Guidance

(January 2019)


ESTABLISH THE BROADEST REASONABLE INTERPRETATION OF THE CLAIM AS A WHOLE
THE STATUTORY CATEGORIES
Step 1
IS THE CLAIM TO A PROCESS, MACHINE, MANUFACTURE OR COMPOSITION OF MATTER?
NO
CAN CLAIM BE AMENDED TO FALL WITHIN A STATUTORY CATEGORY?
YES
WHEN VIEWED AS A WHOLE, THE ELIGIBILITY OF THE CLAIM IS SELF-EVIDENT
CAN ANALYSIS BE STREAMLINED?
A
THE JUDICIAL EXCEPTIONS
Revised Step 2A
IS THE CLAIM DIRECTED TO A LAW OF NATURE, A NATURAL PHENOMENON (PRODUCT OF NATURE) OR ABSTRACT IDEA?
NO
B
YES
IF ABSTRACT IDEA
Prong 1(a)
IDENTIFY SPECIFIC CLAIM LIMITATIONS BELIEVED TO RECITE ABSTRACT IDEA
SUBJECT MATTER GROUPINGS
Prong 1(b)
IS ABSTRACT IDEA A MATH CONCEPT, METHOD OF ORGANIZING HUMAN ACTIVITY, OR MENTAL PROCESS?
NO
ELIGIBLE UNLESS RARE CIRCUMSTANCE
D
TENTATIVE ABSTRACT IDEA: CHECK WITH TC DIRECTOR, JUSTIFY NEW CATEGORY
YES
INTEGRATION / PRACTICAL APPLICATION
Prong 2
DO ADDITIONAL ELEMENTS INTEGRATE THE JUDICIAL EXCEPTION INTO A PRACTICAL APPLICATION
YES
E
NO
THE INVENTIVE CONCEPT
Step 2B
DOES THE CLAIM RECITE ADDITIONAL ELEMENTS THAT AMOUNT TO SIGNIFICANTLY MORE THAN THE JUDICIAL EXCEPTION?
YES
C
NO
CLAIM QUALIFIES AS ELIGIBLE SUBJECT MATTER UNDER 35 USC 101
CLAIM IS NOT ELIGIBLE SUBJECT MATTER UNDER 35 USC 101
A B C D E THE PATHWAYS TO ELIGIBILITY

Subject Matter Eligibility Examples: Abstract Ideas
Example 39 - Method for Training a Neural Network for Facial Detection

Claim:

A computer-implemented method of training a neural network for facial detection comprising:

- collecting a set of digital facial images from a database;
- applying one or more transformations to each digital facial image including mirroring, rotating, smoothing, or contrast reduction to create a modified set of digital facial images;
- creating a first training set comprising the collected set of digital facial images, the modified set of digital facial images, and a set of digital non-facial images;
- training the neural network in a first stage using the first training set;
- creating a second training set for a second stage of training comprising the first training set and digital non-facial images that are incorrectly detected as facial images after the first stage of training; and
- training the neural network in a second stage using the second training set.

Step	Analysis
1: Statutory Category?	**Yes.** The claim recites a series of steps and, therefore, is a process.
2A - Prong 1: Judicial Exception Recited?	**No**. The claim does not recite any of the judicial exceptions enumerated in the 2019 PEG. For instance, the claim does not recite any mathematical relationships, formulas, or calculations. While some of the limitations may be based on mathematical concepts, the mathematical concepts are not recited in the claims. Further, the claim does not recite a mental process because the steps are not practically performed in the human mind. Finally, the claim does not recite any method of organizing human activity such as a fundamental economic concept or managing interactions between people. Thus, the claim is **eligible** because it does not recite a judicial exception.
2A - Prong 2: Integrated into a Practical Application?	**N/A.**
2B: Claim provides an Inventive Concept?	**N/A.**

Patenting AI at the EPO

EPO: The Examination of Computer Implemented Inventions and Artificial Intelligence Inventions

(November 2018)

Mathematical Methods (section G-II, 3.3)

BakerHostetler

- A mathematical method may contribute to the technical character of an invention (i.e. contribute to producing a technical effect that serves a technical purpose) by its **application to a field of technology** and/or by being adapted to a **specific technical implementation.**
- The mere fact that a mathematical method may serve a technical purpose is not sufficient. The claim is to be **functionally limited to the technical purpose**, either explicitly or implicitly.
  - This can be achieved by specifying how the input and the output of the sequence of mathematical steps relate to the technical purpose so that the mathematical method is causally linked to a technical effect.

Mathematical Methods; Technical Applications

BakerHostetler

Examples of technical applications:

- controlling a specific technical system or process, e.g. an X-ray apparatus or a steel cooling process;
- determining from measurements a required number of passes of a compaction machine to achieve a desired material density;
- digital audio, image or video enhancement or analysis, e.g. de-noising, detecting persons in a digital image, estimating the quality of a transmitted digital audio signal;
- separation of sources in speech signals; speech recognition, e.g. mapping a speech input to a text output;
- encoding data for reliable and/or efficient transmission or storage (and corresponding decoding), e.g. error-correction coding of data for transmission over a noisy channel, compression of audio, image, video or sensor data; ….

Mathematical Methods; Technical Applications

Examples (cont'd):

- encrypting/decrypting or signing electronic communications; generating keys in an RSA cryptographic system;
- optimising load distribution in a computer network;
- determining the energy expenditure of a subject by processing data obtained from physiological sensors; deriving the body temperature of a subject from data obtained from an ear temperature detector;
- providing a genotype estimate based on an analysis of DNA samples, as well as providing a confidence interval for this estimate so as to quantify its reliability;
- providing a medical diagnosis by an automated system processing physiological measurements;
- simulating the behaviour of an adequately defined class of technical items, or specific technical processes, under technically relevant conditions.

AI and ML (section G-II, 3.3.1)

BakerHostetler

- AI and ML are based on computational models and algorithms for classification, clustering, regression and dimensionality reduction, such as neural networks, genetic algorithms, support vector machines, k-means, kernel regression and discriminant analysis. Such computational models and algorithms are **per se of an abstract mathematical nature** ... Hence, the guidance provided in G-II, 3.3 generally applies also to such computational models and algorithms.
- The use of a neural network in a heart-monitoring apparatus for the purpose of identifying irregular heartbeats makes a technical contribution.
- The classification of digital images, videos, audio or speech signals based on low-level features (e.g. edges or pixel attributes for images) are further typical technical applications of classification algorithms.
- Classifying text documents solely in respect of their textual content is however **not** regarded to be per se a technical purpose but a linguistic one (T 1358/09).
- Classifying abstract data records or even "telecommunication network data records" without any indication of a technical use being made of the resulting classification is also **not** per se a technical purpose, even if the classification algorithm may be considered to have valuable mathematical properties such as robustness (T 1784/06).
- Where a classification method serves a technical purpose, the steps of generating the training set and training the classifier may also contribute to the technical character of the invention if they support achieving that technical purpose.

Specific technical implementation

BakerHostetler

- *If an algorithm is particularly suitable for being performed on a computer in that its **design was motivated by technical considerations of the internal functioning of the computer**, it may arguably be considered to provide a technical contribution to the invention"*

    - Board of Appeal case T 1358/09

Inventive Step

- ”The presence of an inventive step under Art. 56 EPC requires a non-obvious **technical solution to a technical problem**”
  - *EPO Guidelines for Examination*

- “... where a feature cannot be considered as contributing to the solution of any technical problem by providing a technical effect it has no significance for the purpose of assessing inventive step."
  - *T 0641/00 (Two identities/COMVIK)*

What isn’t technical?

- Business methods
- Aesthetic creations
- Presentations of information
- Mental acts
- Mathematical methods

- Only excluded <u>as such</u>

Reflection point 1

BakerHostetler

- Image classification based on low level features is technical, but text classification per se is ruled out.
- A contradiction? – if a new AI/ML development is applicable to both image and text classification, limitation to image classification seems arbitrary and technically irrelevant.

Reflection point 2

BakerHostetler

- Some AI inventions can relate to **multiple technical use cases**
- How best to protect these?
- Divisional applications? Multiple independent claims? “Or” statement?
- “Specific technical implementation”?

Michael Stein
Mstein@bakerlaw.com
(206) 332-1384