

НЕЙРОСЕТЕВАЯ АСТРОНОМИЯ КАК НОВЫЙ ИНСТРУМЕНТ НАБЛЮДЕНИЯ ЯРКИХ И КОМПАКТНЫХ ОБЪЕКТОВ

А. А. Шацкий^{a*}, И. Ю. Евгеньев^b

^a *Московский государственный университет им. М. В. Ломоносова
119991, Москва, Россия*

^b *Московский авиационный институт
125993, Москва, Россия*

Поступила в редакцию 24 сентября 2018 г.,
после переработки 5 ноября 2018 г.
Принята к публикации 13 ноября 2018 г.

Предложен новый метод решения важной проблемы астрономии, возникающей при наблюдениях на интерферометрах со сверхвысоким угловым разрешением. Этот метод основан на применении теории искусственных нейросетей. Предложена и рассчитана многопараметрическая модель небесного объекта типа Sgr A*. Для этой модели численно построен ряд вероятных изображений для обучения нейросети. После обучения нейросети на этих изображениях качество ее работы проверено на другом ряде изображений из этой же модели. Доказано, что нейросеть может распознавать и классифицировать небесные объекты (получаемые в том числе с интерферометров) практически не хуже, чем это способен делать человек.

DOI: 10.1134/S0044451019040126

1. ВВЕДЕНИЕ

Для наблюдения характеристик объектов, угловые размеры которых сопоставимы с угловыми размерами диаметра горизонта сверхмассивных черных дыр, необходимы интерферометры с угловым разрешением меньше 50 мкс дуги.

В то же время наблюдения многих объектов Вселенной на одиночных телескопах, даже космических, не дают возможности исследовать их структуру ввиду малости угловых размеров. При этом угловое разрешение современных интерферометров с большой базой в оптическом, инфракрасном и радиодиапазонах (последние — так называемые радиointерферометры со сверхдлинными базами, РСДБ) приближается к десяти угловым микросекундам дуги¹⁾ (см. [1–6]).

Речь идет в первую очередь о наблюдениях объектов Sgr A* (угловой диаметр тени $\approx 60 \mu\text{as}$) и ядра M87 (угловой диаметр тени $\approx 40 \mu\text{as}$).

Как известно (см., например, [1]), интерферометры не видят изображений объектов, они видят (измеряют) амплитуду $A(u, v)$ и фазу $\Phi(u, v)$ комплексной функции видности $V(u, v) := A \exp(i\Phi)$ этих объектов:

$$V(u, v) = \iint I(x, y) \exp\left[-\frac{2\pi i(xu + yv)}{\lambda}\right] dx dy, \quad (1)$$

$$I(x, y) = \iint V(u, v) \exp\left[\frac{2\pi i(xu + yv)}{\lambda}\right] du dv. \quad (2)$$

Здесь $I(x, y)$ — функция плотности яркости наблюдаемого объекта в угловых координатах, λ — длина волны, на которой ведется наблюдение. Координаты (u, v) являются фактически двумерными координатами базы интерферометра и называются координатами на (u, v) -плоскости.

Наблюдения на сверхмалых угловых масштабах могут быть выполнены только интерферометрами со сверхбольшими базами: более $5 \cdot 10^9 \lambda$, см. [1]. Главной проблемой здесь является измерение фазы при достаточно большой разнице путей до телескопов интерферометра. Минимально необходимой точностью при этом является четверть длины волны $\lambda/4$, что соответствует фазе, измеренной с точностью, много меньшей $\pi/2$. Для интерферометров в инфракрасном и оптическом диапазонах данное

* E-mail: shatskiyalex@gmail.com

¹⁾ Угловая микросекунда дуги $1 \mu\text{as} \approx 4.8 \cdot 10^{-12}$ рад.

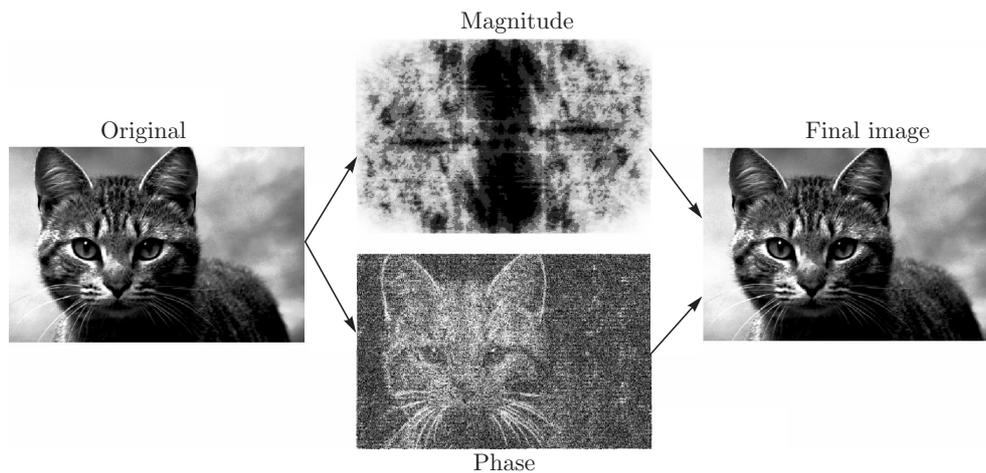


Рис. 1. Прямое и обратное преобразование Фурье для изображения кота. Изображение для амплитуды (Magnitude) получено как восстановленное (обратным преобразованием Фурье) изображение с фазой равной единице и амплитудой кота, а изображение для фазы (Phase) получено как восстановленное изображение с амплитудой равной единице и фазой кота

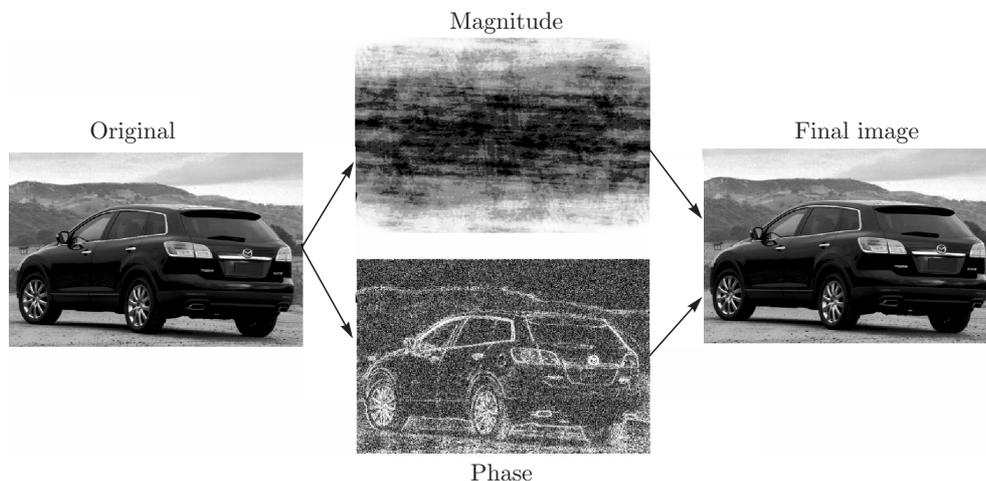


Рис. 2. То же, что на рис. 1, но для изображения автомобиля

требование является, в принципе, выполнимым, но спектральная разрешающая способность телескопов ($\lambda/\Delta\lambda$) в таких интерферометрах оказывается порядка или меньше единицы. Это приводит к сильному размытию интерференционной картины. Поэтому без точного знания спектров деталей изображения обработка этого сигнала бессмысленна в принимаемом диапазоне $\Delta\lambda$. К сожалению, мы можем измерить лишь суммарный спектр всего изображения, но не можем измерить спектры деталей этого изображения. Поэтому количественно обработать интерферометрические данные мы можем только в тех диапазонах, где допустимо монохромное приближение для спектральной разрешающей способности: $\lambda/\Delta\lambda \gg 1$.

На сегодняшний день это соответствует телескопам, работающим на длинах волн больше ~ 1 мм.

2. ОСОБЕННОСТИ ИНТЕРФЕРЕНЦИОННОЙ АСТРОНОМИИ

К сожалению, большинство интерферометров со сверхвысоким угловым разрешением (т.е. со сверхбольшими базами) не могут измерять фазу функции видности, они регистрируют только ее амплитуду. В русскоязычной литературе вместо слова амплитуда принято использовать термин «автокорреляционная функция», а в англоязычной — «correlated flux density», далее везде мы будем использовать сокращение CFD. Фаза функции видности безвозврат-

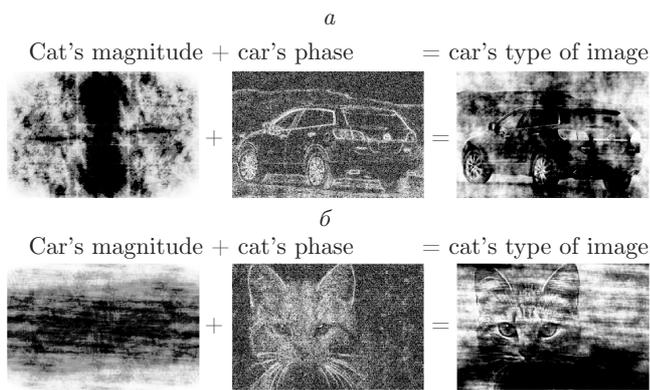


Рис. 3. а) Амплитуда кота + фаза автомобиля, таким образом восстановленное изображение похоже на автомобиль. б) Амплитуда автомобиля + фаза кота, таким образом восстановленное изображение похоже на кота

но теряется при превышении длины корреляции, а также из-за атмосферных (и других) помех. Тем не менее, даже знание одной только функции CFD может многое прояснить об изображении объекта и его свойствах.

Что можно сказать об изображении без знания фазы? На эту тему написано немало работ (см., например, [7, 8]).

На рис. 1–3 продемонстрированы прямые и обратные преобразования Фурье для двух изображений (в оттенках серого): кота и автомобиля. Как понятно из самих рисунков и подписей к ним, основная информация, позволяющая человеку отождествить изображение, находится в фазе Φ . Но это еще не означает, что в CFD содержится меньше информации, чем в фазе Φ . При этом по каким-то причинам человеческий мозг и его зрение не в состоянии различать информацию, содержащуюся в CFD. Человек различает только информацию, содержащуюся в фазе Φ .

Зная только CFD, мы до недавнего времени имели единственную возможность определять характеристики изображения — сравнивать распределение наблюдаемой CFD с ее модельными значениями в виде некоторой двумерной функции, зависящей от определенного набора параметров. Таким образом, мы могли выбирать наилучшее модельное изображение, соответствующее определенному набору этих параметров (наилучшему вектору в пространстве параметров). Существует много математических критериев степени сходства (и различия) функций. Одним из них является известный критерий минимальности хи-квадрат (χ^2):

$$\chi^2 = \sum_i \frac{(A_{model,i} - A_{obs,i})^2}{\Delta A_{obs,i}^2}. \quad (3)$$

Здесь индексы «model» и «obs» отвечают соответственно модельной и наблюдаемой амплитудам, суммирование ведется по всем точкам изображения, данные о которых имеются в распоряжении наблюдателя.

Критерий минимальности хи-квадрат является достаточно простым и надежным, однако при большом числе параметров модели он требует экспоненциально большого времени счета.

Допустим, мы имеем N -параметрическую модель CFD как двумерную функцию на (u, v) -плоскости.

Пусть также имеются T наблюдательных точек на (u, v) -плоскости, в которых мы будем сравнивать результаты наблюдений и модели. В N -мерном параметрическом пространстве мы должны выбрать куб и каждое ребро этого куба разбить на K частей (в принципе, это число может быть разным для каждого ребра). Тогда полное число точек (ячеек), в которых мы должны вычислить величину хи-квадрат, получается равным K^N , а для вычисления хи-квадрат мы должны умножить K^N на $5T$, чтобы получить полное число необходимых арифметических операций. Реально же нам потребуется еще больше вычислительных ресурсов, так как каждая модельная точка обычно представляет собой арифметическое выражение из специальных функций (обычно функций Бесселя) и их вычисление также требует времени. Например, полное число арифметических операций X для $N = 5$, $K = 100$ и $T = 100$ получается равным $X = 5 \cdot 10^{12}$ (не считая вычислений самих модельных точек). Этот пример мы выбрали не случайно: он соответствует готовому расчету моделей астрофизиками из группы проекта телескопа горизонта событий [9].

Возникает вопрос: почему нельзя использовать стандартные методы нахождения глобального минимума N -мерной функции в N -мерном пространстве параметров?

К сожалению, прямые и очевидные методы оптимизации таких вычислений, например метод градиентного спуска в функционале (3), в данном случае неприменим. Это связано с тем, что вычисление такого рода стандартными методами приводит к одному из локальных минимумов. А нам необходим глобальный минимум (или локальный, но рядом с глобальным). Поскольку при больших N локальных минимумов становится экспоненциально много, шансов найти среди них глобальный становится



Рис. 4. Модель «размытого» (по Гауссу) полумесяца, хорошо аппроксимирующая тень от черной дыры

ся чрезвычайно мало. Также стоит отметить, что, несмотря на неприменимость метода градиентного спуска к нахождению минимума функционала (3), этот метод повсеместно применяется при обучении нейронных сетей — алгоритм обратного распространения ошибки (о котором будет сказано ниже) как раз позволяет эффективно вычислить градиент функции потерь по весам.

В работе [10] были исследованы свойства CFD для объектов, которые могут являться черными дырами. Общим для моделей черных дыр являются тени, которые гравитация черной дыры оставляет на фоне излучения, источником которого является материя, окружающая черную дыру. Источником этого излучения может быть материя, составляющая аккреционный диск вокруг черной дыры, или это излучение просто является фоновым, в том числе реликтовым микроволновым излучением. В любом случае форма этой тени на картине излучения, регистрируемого наблюдателем, весьма характерна. Очень хорошим приближением для этой формы является модель полумесяца, или модель «размытого» по Гауссу (см. рис. 4) полумесяца (см., например, [9]).

Модель полумесяца весьма проста и эффективна для численного моделирования. В ней всего пять числовых параметров и, как сказано выше, число арифметических операций для этой модели примерно равно 10^{13} . Для современных компьютеров это посильная задача.

Но что делать, если наша модель сложнее и имеет, например, 11 параметров? В этом случае (при $K = 100$ и $T = 100$) число арифметических операций достигает величины $\sim 10^{25}$. Это уже запредельная величина даже для современных суперкомпьютеров. А ведь число параметров N в более реальных моделях может быть гораздо большим (около сотни).

Что делать в этом случае?

3. НЕЙРОСЕТЕВАЯ АСТРОНОМИЯ

Наука о нейросетевой астрономии является совершенно новой, и данное определение было введено одним из авторов этой работы. Сначала поясним, что такое искусственные нейронные сети (или сокращенно нейросети), о которых далее пойдет речь.

Искусственная нейронная сеть — это математическая модель (а также ее программное или аппаратное воплощение), построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке моделировать эти процессы.

Практически решаемыми задачами для многих (искусственных) нейросетей являются задачи прогнозирования, аппроксимации, кластеризации, сжатия данных, локализации объектов, а также разбиение заданного множества объектов на классы (а иногда еще и на подклассы). Нас будет интересовать именно разбиение заданного множества объектов на классы, т. е., кратко говоря, классификация множества. С этой целью проводится обучение нейросети — на ее вход по очереди подаются объекты, выбранные случайным образом из обучающего множества. На выходе нейросеть выдает вектор классификатора. В идеале для совершенно обученной нейросети этот вектор заполнен нулями и одной единицей — номер позиции этой единицы соответствует номеру класса текущего объекта на входе. В реальности же этот вектор состоит из нецелых чисел (float) и нормирован на единицу. Из этого вектора можно получить величину ошибки, с которой нейросеть определила искомый класс. С помощью алгоритма «обратного распространения ошибки» эта ошибка распространяется от конца сети к ее началу и в процессе этого происходит коррекция весовых коэффициентов нейросети (которые и составляют ее основу). Этот процесс повторяют до тех пор, пока

ошибка на выходе не становится достаточно малой, тогда нейросеть готова к работе.

Подробнее про искусственные нейронные сети можно прочитать в работах [11–14]. Про нейросети, используемые в настоящей работе, можно прочитать в цикле статей [15–21], а также в [22].

Как упоминалось выше, по каким-то причинам человеческий мозг и его зрение не в состоянии различать информацию, содержащуюся в CFD, хотя там также содержится достаточно много информации об объекте. Идея данной работы состоит в том, чтобы обучить нейросеть на объектах, являющихся амплитудами фурье-образов изображений, т. е. CFD. Как будет видно ниже, это оказывается вполне решаемой задачей.

Таким образом, можно обучить нейросеть на разных классах в выбранной модели — для широкого круга протяженных астрономических объектов или группы точечных объектов.

Обучить нейросеть можно одинаково хорошо вне зависимости от сложности моделей этих объектов. Эта уникальная возможность решает проблему, сформулированную в предыдущем разделе.

4. МОДЕЛЬ ДЛЯ НЕЙРОСЕТИ

Для данной работы мы выбрали относительно простую модель, содержащую 11 параметров. Тем не менее, как упомянуто выше, даже с помощью суперкомпьютеров эта модель не может быть рассчитана стандартными методами. Мы покажем, что нейросети способны классифицировать такую модель с вероятностью, близкой к 100%.

В основу модели положим модель полумесяца — это пять параметров: радиус внешнего круга, радиус внутреннего круга, две координаты центра внутреннего круга (относительно центра внешнего круга

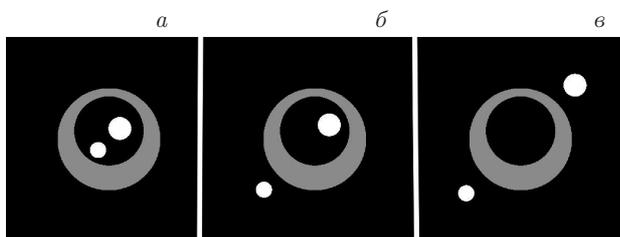


Рис. 5. 11-параметрическая модель для обучения нейросети на трех классах: *a* — нулевой класс (нет звезд вне круга полумесяца), *b* — первый класс (одна звезда вне круга полумесяца), *c* — второй класс (две звезды вне круга полумесяца)

га) и яркость полумесяца. Кроме того, добавляем к модели еще два точечных объекта²⁾ (две звезды), у каждого из которых по три параметра: координаты и яркость (рис. 5).

При этом согласно исследованию, проведенному в работе [10], звезды внутри внешнего круга полумесяца соответствуют модели черно-белой дыры, или кротовой норы, а звезды вне внешнего круга полумесяца соответствуют просто звездам на фоне черной дыры (или звездам гравитационно-линзированным этой черной дырой).

Обучение можно провести как на оригинальных изображениях модели (которых в реальности мы не имеем, как было показано в разд. 2), так и на изображениях CFD, соответствующих оригинальным изображениям.

Кроме того, мы также можем попытаться выполнить обратные фурье-преобразования для изображений CFD, взяв при этом вместо отсутствующей фазы постоянную двумерную функцию, и получить в результате еще один набор изображений (также разбитых на классы), а после этого обучить нейросеть на этом новом наборе изображений.

5. ОБУЧЕНИЕ НЕЙРОСЕТИ

Мы выбрали для обучения нейросеть YOLOv3 из пакета DARKNET [22] (см. Приложение). Этот пакет относится к числу открытого программного обеспечения с лицензией BSD. Такой выбор обусловлен тем, что, во-первых, данная нейросеть достаточно качественно может различать классы, на которых была обучена. Во-вторых, эта нейросеть может также строить рамки вокруг найденных на изображении объектов — это может потребоваться для идентификации подклассов (подмножеств) на множествах объектов (в дальнейших исследованиях). В-третьих, эта нейросеть также выдает вероятность определенного ей объекта (обведенного рамкой, с присвоением ей идентификатора), таким образом мы можем отсекавать те объекты, вероятность обнаружения которых недостаточно велика для нас.

²⁾ В самой модели мы добавляем на картинки не яркие точки, а яркие кружки, радиус которых пропорционален модельной яркости звезды, так как численное моделирование точечных объектов дает неадекватные результаты по сравнению с моделированием протяженных объектов. Кроме того, в реальных телескопах звезды дают изображения именно в виде кружков (а не точек), причем чем ярче звезда, тем больше кружок. А в дальнейших фурье-преобразованиях мы моделируем эти же звезды именно как точки (дельта-функции).

Мы также пытались обучить гораздо более простые (неглубокие) нейросети на этом же множестве изображений. Эти неглубокие нейросети представляли собой всего три сверточных слоя и один полносвязный (перцептрон) слой на выходе. Однако данная конфигурация не позволила получить вероятность распознавания классов выше 70 %.

Качество обучения нейросети мы решили проверить для трех разных наборов картинок: 1) для набора оригинальных картинок модели, 2) для набора CFD-картинок, полученных аналитическим фурье-преобразованием из оригинальных (с учетом использования дельта-функции в фурье-преобразовании для моделей звезд), 3) для набора картинок, полученных из CFD-картинок путем численного обратного фурье-преобразования с использованием постоянной функции фазы (далее для краткости CFD2DFT-картинки).

Для аналитического получения CFD-картинок по заданным 11 параметрам мы использовали формулу для CFD из работы [10]:

$$A_{crescent+points}(u, v) = \left[\frac{B_0}{\eta} \left[r_{out} J_1(\eta r_{out}) - \exp\left(-\frac{2\pi i(x_c u + y_c v)}{\lambda}\right) \times r_{in} J_1(\eta r_{in}) \right] + \sum_j B_j \exp\left(-\frac{2\pi i(x_j u + y_j v)}{\lambda}\right) \right]. \quad (4)$$

Здесь $\eta := \sqrt{u^2 + v^2}$, B_0 — плотность яркости полумесяца, r_{in} и r_{out} — его внутренний и внешний радиусы, x_c и y_c — координаты центра внутреннего круга полумесяца (относительно внешнего), B_j — яркость звезды с индексом j , x_j и y_j — ее координаты, J_1 — функция Бесселя первого порядка, λ — длина волны, на которой ведутся наблюдения.

Все картинки и преобразования над ними мы подготовили с использованием пакета OpenCV на C++.

Для каждого из этих трех наборов картинок мы подготовили обучающее множество из 3000 картинок — по 1000 картинок для каждого из трех классов (0, 1, 2). На каждом из этих наборов (по 3000 картинок) нейросеть проходила обучение.

Кроме того, мы подготовили еще по 600 картинок для каждого набора (по 200 картинок в каждом классе) для тестов обученных нейросетей. Все эти картинки по 11 параметрам модели случайным образом отличаются друг от друга. Кроме того, мы проследили, чтобы в картинках две звезды не накладывались друг на друга (слишком близко) и не

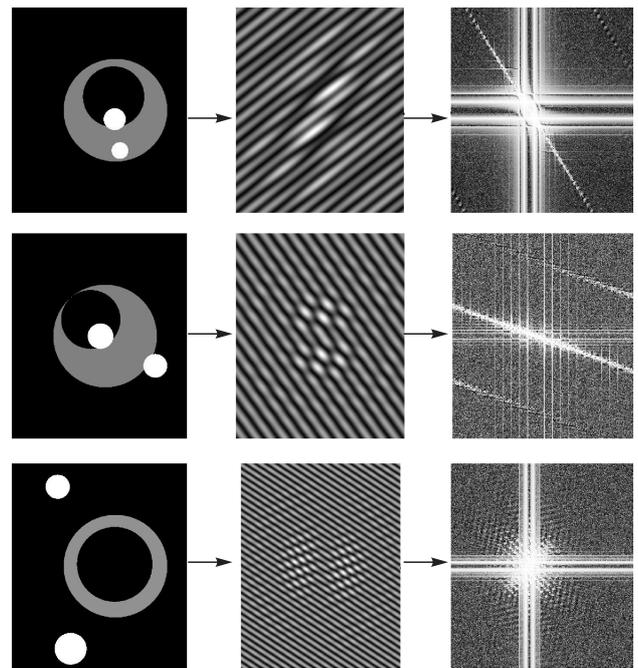


Рис. 6. Слева расположены оригинальные картинки, в центре — CFD-картинки, полученные с теми же 11 параметрами, что и оригинальные, справа — картинки, полученные из CFD-картинок путем обратного фурье-преобразования с использованием постоянной функции фазы (CFD2DFT-картинки). Верхний ряд — для нулевого класса, средний — для первого, нижний — для второго

лежали слишком близко к границе внешнего круга полумесяца, чтобы классы явно отличались друг от друга.

Примеры таких картинок (для каждого из трех классов) представлены на рис. 6.

6. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ НЕЙРОСЕТИ

В числе прочих характеристик и параметров (о которых будет сказано ниже) нейросеть выдает вероятность «prob» распознавания классифицируемого ей объекта (см. Приложение). Обозначим $prob_{min}$ минимальный порог для этой вероятности.

Далее фигурирует еще одна вероятность правильного распознавания классов, определяемая как часть правильных ответов среди всех ответов нейросети на выборке из 600 тестовых картинок (по 200 картинок в каждом классе), обозначим эту вероятность символом ρ .

При $prob_{min} = 0.1$ вероятности ρ получились следующими:

- для оригинальных картинок $\rho \approx 98.83\%$,
- для CFD-картинок $\rho \approx 97.67\%$,
- для CFD2DFT-картинок $\rho \approx 92.20\%$.

При этом все картинки были распознаны (либо правильно, либо нет), т.е. величина prob для всех шести сотен картинок оказалась больше, чем prob_{\min} . Анализ ошибок показал, что нейросеть ошибалась только в случаях, когда две звезды частично накладывались друг на друга, т.е. нейросеть принимала их за одну звезду (а вторую найти не могла).

Если же мы хотим отбросить недостоверные результаты, то нужно повысить минимальный порог prob_{\min} .

При $\text{prob}_{\min} = 0.9$ вероятности ϱ получились следующими:

- для оригинальных картинок $\varrho \approx 99.32\%$, 15 картинок были нераспознаны;
- для CFD-картинок $\varrho \approx 98.97\%$, 20 картинок были нераспознаны;
- для CFD2DFT-картинок $\varrho \approx 95.19\%$, 122 картинки были нераспознаны.

При $\text{prob}_{\min} = 0.95$ вероятности ϱ получились следующими:

- для оригинальных картинок $\varrho \approx 99.30\%$, 28 картинок были нераспознаны;
- для CFD-картинок $\varrho \approx 99.13\%$, 28 картинок были нераспознаны;
- для CFD2DFT-картинок $\varrho \approx 95.57\%$, 171 картинка была нераспознана.

7. ДОБАВЛЕНИЕ ШУМОВ НА КАРТИНКИ

При реальных измерениях на интерферометрах у наблюдателя нет полностью заполненной (u, v) -плоскости, имеются лишь отдельные кривые линии на этой плоскости. Поэтому, чтобы получить результат, незаполненную часть (u, v) -плоскости аппроксимируют на основе имеющихся наблюдательных данных (линий на (u, v) -плоскости). При этом неизбежно возникают ошибки, связанные с этой аппроксимацией. Кроме того, добавляются ошибки, связанные с шумами, накладывающимися на данные наблюдений.

Для того чтобы оценить влияние всех этих ошибок и шумов на наши результаты, мы искусственно «испортили» сгенерированные CFD-картинки. Сделали мы это, наложив на картинки blur-фильтр, который делает «размытие» (или сглаживание) картинки с заданным параметром области «размытия». При этом нейросеть на «размытых» картинках не переобучалась.

Размеры всех наших картинок составляли 480×480 пикселей.

При размерах области «размытия» 3×3 пикселей:

- для $\text{prob}_{\min} = 0.1$ имеем $\varrho \approx 98.17\%$, все картинки были распознаны;
- для $\text{prob}_{\min} = 0.9$ имеем $\varrho \approx 98.80\%$, 15 картинок были нераспознаны;
- для $\text{prob}_{\min} = 0.95$ имеем $\varrho \approx 99.13\%$, 27 картинок были нераспознаны.

При размерах области «размытия» 5×5 пикселей:

- для $\text{prob}_{\min} = 0.1$ имеем $\varrho \approx 96.83\%$, все картинки были распознаны;
- для $\text{prob}_{\min} = 0.9$ имеем $\varrho \approx 98.78\%$, 27 картинок были нераспознаны;
- для $\text{prob}_{\min} = 0.95$ имеем $\varrho \approx 99.10\%$, 42 картинки были нераспознаны.

При размерах области «размытия» 10×10 пикселей:

- для $\text{prob}_{\min} = 0.1$ имеем $\varrho \approx 93.67\%$, все картинки были распознаны;
- для $\text{prob}_{\min} = 0.9$ имеем $\varrho \approx 96.55\%$, 50 картинок были нераспознаны;
- для $\text{prob}_{\min} = 0.95$ имеем $\varrho \approx 96.82\%$, 65 картинок были нераспознаны.

При размерах области «размытия» 20×20 пикселей:

- для $\text{prob}_{\min} = 0.1$ имеем $\varrho \approx 65.61\%$, 1 картинка была нераспознана;
- для $\text{prob}_{\min} = 0.9$ имеем $\varrho \approx 71.69\%$, 155 картинок были нераспознаны;
- для $\text{prob}_{\min} = 0.95$ имеем $\varrho \approx 74.18\%$, 205 картинок были нераспознаны.

Отсюда видно, что данное «размытие» (или сглаживание) картинок, как и следовало ожидать, отрицательно влияет на результаты правильного распознавания классов нашей нейросетью. Но итоговые результаты все равно оказываются вполне приемлемыми для использования нейросетей в качестве искусственного интеллекта распознавания астрономических объектов. Мы предполагаем, что столь незначительное влияние шумов на результаты связано с тем, что CFD-картинки уже по сути являются неким интегральным «размытием» оригинальных картинок, так как фурье-преобразование и есть такое интегральное «размытие»: каждая точка в CFD-картинке содержит в себе информацию, получаемую от всех точек в оригинальной картинке.

8. ОБСУЖДЕНИЕ

Как было доказано в предыдущем разделе, использование нейросетей для определения типов объектов CFD дает очень хорошие результаты: вероятность распознавания при классификации CFD-картинок близка к 100%.

Кроме вероятности распознавания prob классифицируемого объекта, нейросеть выдает еще ряд прочих полезных параметров: координаты рамки, ограничивающей определенный на картинке объект, идентификатор этой рамки, необходимый для выделения и дальнейшего сопровождения деталей объекта. Поэтому, если вероятность распознавания окажется недостаточно высокой, это будет означать, что используемая модель недостаточно полно и хорошо описывает входящую в нейросеть картинку, т. е. нужно будет расширить и усложнить модель, а затем переобучить нейросеть. С другой стороны, если мы получим достаточно высокую вероятность классифицируемого объекта, то модель для этого объекта можно будет расширить: разбить имеющиеся классы на дополнительные подклассы и переобучить нейросеть на новом наборе классов и подклассов. Таким образом можно будет выявить на входящей картинке отдельные детали объекта (класса), соответствующие подклассам объекта со сложной структурой. И так далее с новыми подклассами. Ничего подобного стандартные методы изучения CFD не позволяют сделать.

Таким образом, использование нейросетей дает гораздо более надежные результаты, чем стандартные методы моделирования по нескольким параметрам (где вероятность ошибки обычно в несколько раз больше, чем у нейросетей). Но главное — использование нейросетей не ограничено количеством параметров в модели. То есть практически с тем же успехом можно использовать сколь угодно сложную (многопараметрическую) модель оригинальной картинки (и соответственно CFD-картинки тоже).

Результаты этой работы могут оказаться полезны исследователям международного проекта Event Horizon Telescope: <https://eventhorizontelescope.org>.

ПРИЛОЖЕНИЕ

Архитектура нейросети YOLOv3

Архитектура сети YOLOv3 основана на Darknet-53 (цифра в названии говорит о количестве сверточных слоев в архитектуре). Архитектура использует остаточные (residual) блоки и повышение дис-

кретизации (upsampling) и по сути делает детектирование в трех масштабах, деля изображение на ячейки-клетки 13×13 , 26×26 , 52×52 . На выходе мы получаем три тензора $13 \times 13 \times (B \times (5 + C))$, $26 \times 26 \times (B \times (5 + C))$, $52 \times 52 \times (B \times (5 + C))$, где B — количество рамок, центр которых находится в ячейке, при этом размеры рамки могут выходить за границу ячейки, C — вероятности классов из датасета (их сумма равна единице), цифра 5 обозначает пять параметров: вероятность P того, что в данной ячейке обнаружен объект, координаты рамки x, y , ее высота и ширина w, h . Количество рамок B , большее одной, необходимо в том числе для таких ситуаций, когда в одной ячейке находятся центры двух объектов (например, пешеход на фоне машины). Например, для $B = 2$ и $C = 3$ для каждой ячейки-клетки в трех масштабах получим $3549 (13 \cdot 13 + 26 \cdot 26 + 52 \cdot 52)$ векторов $P1, x1, y1, w1, h1, C1_1, C1_2, C1_3, P2, x2, y2, w2, h2, C2_1, C2_2, C2_3$, из которых впоследствии отфильтруем только нужные нам. В архитектуре YOLOv3 $B = 3$. Возвращаясь к вопросу о вероятности prob , которую выдает программа для каждого найденного ей объекта, это вероятность P , умноженная на C того класса, чей тип вероятности (score probability) $(C1_1, C1_2, C1_3)$ максимален для данной рамки.

ЛИТЕРАТУРА

1. A. R. Thompson, J. M. Moran, and G. W. Swenson, Jr., *Interferometry and Synthesis in Radio Astronomy*, Wiley (2001).
2. K. I. Kellermann and J. M. Moran, *Ann. Rev. Astron. Astrophys.* **39**, 457 (2001).
3. A. Quirrenbach, *Ann. Rev. Astron. Astrophys.* **39**, 353 (2001).
4. A. Quirrenbach, *ISSI Sci. Rep. Ser.* **9**, 293 (2010).
5. R.-S. Lu, A. E. Broderick, F. Baron, J. D. Monnier, V. L. Fish, S. S. Doeleman, and V. Pankratius, *Astrophys. J.* **788**, 120 (2014).
6. Н. С. Кардашев и др., *Астрон. ж.* **90**, 179 (2013).
7. P. Coles and L.-Y. Chiang, *Nature* **406**, 376 (2000).
8. L.-Y. Chiang et al., *Astrophys. J.* **590**, L65 (2003).
9. A. B. Kamruddin and J. Dexter, *Month. Not. Roy. Astron. Soc.* **434**, 765 (2013).
10. А. А. Шацкий, Ю. Ю. Ковалев, И. Д. Новиков, *ЖЭТФ* **147**, 926 (2015).

11. В. В. Круглов, В. В. Борисов, *Искусственные нейронные сети. Теория и практика*, Горячая линия. Телеком, Москва (2002).
12. R. Hadsell, S. Chopra, and Y. LeCun, *Dimensionality Reduction by Learning an Invariant Mapping* (2006), <http://yann.lecun.com/exdb/publis/pdf/hadsell-chopra-lecun-06.pdf>.
13. К. В. Воронцов, *Нейронные сети, видеокурс* (2016), <https://www.youtube.com/watch?v=WjwA5DqxL-c>.
14. F.-F. Li, J. Johnson, and S. Yeung, *Convolutional Neural Networks* (2018), <http://cs231n.stanford.edu/syllabus.html>.
15. K. Ehsani, H. Bagherinezhad, J. Redmon, R. Motlaghi, and A. Farhadi, arXiv:1803.10827.
16. J. Redmon and A. Farhadi, arXiv:1804.02767.
17. D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, arXiv:1712.03316.
18. J. Redmon and A. Farhadi, arXiv:1612.08242.
19. M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, arXiv:1603.05279.
20. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, arXiv:1506.02640.
21. J. Redmon and A. Angelova, arXiv:1412.3128.
22. A. Bochkovskiy, <https://github.com/AlexeyAB/darknet>.